U N I V E R S I D A D D E G U A Y A Q U I L

F a c u l t a d d e C i e n c i a s M a t e m á t i c a s y F í s i c a

C a r r e r a d e I n g e n i e r í a e n S i s t e m a s

C o m p u t a c i o n a l e s

“S e r v i d o r d e A r c h i v o s C o r p o r a t i v o”

T O M O I

P R O Y E C T O D E G R A D O

P r e v i o a l a O b t e n c i ó n d e l T í t u l o d e :

I N G E N I E R O E N S I S T E M A S C O M P U T A C I O N A L E S

A u t o r e s :

H e r n á n J a r a m i l l o J i m é n e z

J o r g e R e y e s T o m a l á

M ó n i c a A z u c e n a Z ú ñ i g a T a p i a

G U A Y A Q U I L – E C U A D O R

A ñ o : 2 0 0 8

AGRADECIMIENTO

Agradezco de todo corazón a Jehová, por su guía y bendición y por haber permitido que llegue a la culminación de mis estudios, también a mi madre por su apoyo constante y a mis compañeros Hernán y Jorge por el compañerismo y el esfuerzo que entregaron para la realización de este proyecto.

Mónica

Agradezco a Dios por la orientación y protección que me ha dado, por permitirme entregarle este logro que he conseguido a un en vida a mis padres ya que gracias a la enseñanza escondida que me han dado, al demostrarme la fuerza que tienen para poder salir de las adversidades de la vida.

Hernán

Agradecimiento enorme a mi Familia y a todas las personas que han colaborado para cumplir este objetivo.

Jorge

DEDICATORIA

Dedico este proyecto que representa una larga trayectoria de mi vida llena de esfuerzo, dedicación y estudios a mi madre que es la persona que ha estado siempre a mi lado con la ayuda de Jehová apoyándome día a día y este es el resultado que siempre soñé con entregarle como premio a su amor y dedicación desinteresada.

Mónica

Dedico este proyecto a mis dos preciosas gardenias Sasha y Damaris. A mi querida esposa Yanine, a mi madre que siempre estuvieron dándome animo para seguir adelante.

Hernán

Dedicada a mi Familia, y a todas las personas que han colaborado para cumplir este objetivo.

Jorge

TRIBUNAL DE GRADUACIÓN

Presidente del Tribunal	Primer Vocal
Segundo Vocal	Secretario

DECLARACIÓN EXPRESA

"La autoría de la tesis de grado corresponde exclusivamente al suscrito(s), perteneciendo a la Universidad de Guayaquil los derechos que generen la aplicación de la misma".

(Reglamento de Graduación de la Carrera de Ingeniería en sistemas Computacionales, Art. 26)

Hernán Jaramillo Jiménez
0919367730

Jorge Reyes Tómala
0914388210

Mónica Zúñiga
0920337136

RESUMEN

El Servidor de Archivos Corporativo fue desarrollado para centralizar la información y poderla compartir entre los usuarios de una red, conservando la integridad de la misma, logrando que ésta se encuentre disponible y actualizada en todo momento. La aplicación se basa en la comunicación de dos plataformas diferentes como Linux actuando como File Server y clientes Windows XP, para lo cual se empleó la distribución CentOS 5.0 y como herramienta de desarrollo JDevstudio10132. Para comunicarse con la aplicación es necesario un código de usuario y su respectiva clave de acceso a través de un Web browser.

INDICE GENERAL

INDICE GENERAL 1

CAPITULO 1

SERVIDOR DE ARCHIVOS CORPORATIVOS 1

1.1 Antecedentes 1

1.2 Problemática 2

1.3 Solución 3

1.4 Definición de un Servidor de Archivos 4

1.5 Visión 5

1.6 Misión 5

1.7 Objetivos 6

1.7.1 General 6

1.7.2 Específicos 6-7

1.8 Alcances 8-9

1.9 Identificación de la alternativa de la solución 10

1.10 Recursos 10

1.10.1 Recursos hardware 10

1.10.2 Recursos Software 12

1.10.2.1 Recursos Software Servidor 12

1.10.2.2 Recursos Software Clientes 12

1.10.3 Recursos Humanos 13

1.11 Metodología 13

1.12 Cronograma de Actividades 13-15

CAPITULO 2

ANALISIS 17

2.1 Levantamiento de la Información 18

2.1.1 Entrevista 18

2.1.2 Investigación 18

2.2 Análisis de los requerimientos 19

2.2.1 Análisis de Recursos Humanos 19

2.2.2 Análisis de Recursos de Hardware y Software 19

2.3 Estudio de Factibilidad 20

2.3.1 Análisis Beneficio-Costo 20

2.3.1.1 Factibilidad Técnica 20

2.3.1.2 Factibilidad Operacional 21

2.3.1.3 Factibilidad Económica 22

2.4 Diccionario de Datos 22-23

2.5 Acerca de Samba 24

2.6 Configuración del Servicio Samba 24

2.7 Arquitectura Samba 25

2.8 Caso de Uso 25-28

2.9 Diagrama de Flujo 29-38

CAPITULO 3

DISEÑO 40

3.1 Diseño de Datos 41

3.1.1 Estructura de Datos 42

3.1.1.1 Tablas 42

3.1.1.2 Claves Primarias 43

3.1.1.3 Claves Foráneas 43

3.2 Diseño Arquitectónico 43

3.3 Diseño de Interfaz 44

3.3.1 Interfaz Principal 45-58

CAPITULO 4

DESARROLLO Y PRUEBAS DEL SISTEMA 50

4.1 Configuración del servidor de Archivos Corporativo 51

4.1.1 Instalación, Configuración del DNS 51

4.1.1.1 Pruebas del servidor DNS 52

4.1.2 Instalación, Configuración de samba como PDC 52-54

4.1.3 Instalación, Configuración OpenLDAP 54-58

4.1.4 Integración Samba con OpenLDAP 58-60

4.1.5 Instalación, Configuración de SMBLDAP-TOOLS 60-63

4.1.6 Instalación, Configuración servidor VSFTP 63-64

4.1.7 Asignación de Cuotas 64

4.1.8 Configuración servidor de correo sendmail 65-67

4.1.9 configuración servidor Web HTTP 67-69

4.2 Desarrollo e implementación Base Datos 69

4.2.1Instalacion MySQL 69

4.2.2 Creación de la base de Datos 70-72

4.2.2.1 Conexión a la base de Datos 73

4.3 Desarrollo de Aplicación Web Para la Administración del proyecto. 74-75

4.3.1 Clases. **75**

4.3.2 Scripts **76-79**

CAPITULO 5

5 Implementación del Sistema 80

5.1 Introducción **81**

5.2 Elementos Físicos **82**

5.3 Elementos Lógicos **82**

5.4 Elementos Humanos **82**

5.5 Infraestructura **83**

CAPITULO 6

6 Recomendaciones y Conclusiones 85

6.1 Recomendaciones **86**

6.2 Hardware **86**

6.3 Software **87**

6.4 Cableado **88**

6.5 Puesta en Marcha **88**

6.6 Conclusiones **88**

6.7 Glosario de Términos **89-94**

Anexo 1 Cronograma de Actividades **95**

Anexos 2 Archivos de Configuraciones **96**

2.1 DNS

2.1.1 sacs.com.zone **97**

2.1.2 15.168.192.a **97**

2.1.3 named.conf **98**

2.1.4 resolv.conf **98**

2.2 samba **99**

2.2.1 smb.conf **99-104**

2.3 openldap **105**

2.3.1 sldap.conf **105-106**

2.4 smbldap-tools **107**

2.4.1 smbldap_bind.conf **107**

2.4.2 smbldap.conf **107-110**

INDICE GRAFICO

Figura 1.1 Arquitectura del servidor de archivos 5

Figura 1.2 Interfaz Web de la aplicación 9

Tabla 2.1 Gastos de Recursos de Hardware 19

Tabla 2.2 Gastos de Recursos de Software 19

Tabla 2.3 Gastos de Recursos de Software 19

Figura 2.1 Arquitectura Samba 23

Figura 2.2 Caso de Uso de forma General 24

Figura 2.3 Escenario de Usuarios de Red 25

Figura 2.4 escenario del usuario Administrador 26

Figura 3.1 Diseño de Datos 39

Figura 3.2 Diseño Arquitectónico 41

Figura 3.2 Diseño de la Interfaz 45

Figura 3.3 Creadores 45

Figura 3.4 Misión 46

Figura 3.5 Vision 46

Figura 3.6 Menú Administrador 47

Figura 3.7 Grupos 47

Figura 3.8 Menú Archivos 48

Figura 3.9 Menú Usuario 48

CAPITULO 1

SERVIDOR DE ARCHIVOS CORPORATIVOS 1

1.1 Antecedentes 1

1.2 Problemática 2

1.3 Solución 3

1.4 Definición de un Servidor de Archivos 4

1.5 Visión 5

1.6 Misión 5

1.7 Objetivos 6

1.7.1 General 6

1.7.2 Específicos 6-7

1.8 Alcances 8-9

1.9 Identificación de la alternativa de la solución 10

1.10 Recursos 10

1.10.1 Recursos hardware 10

1.10.2 Recursos Software 12

1.10.2.1 Recursos Software Servidor 12

1.10.2.2 Recursos Software Clientes 12

1.10.3 Recursos Humanos 13

1.11 Metodología 13

1.12 Cronograma de Actividades 13-15

CAPITULO 1

1 SERVIDOR DE ARCHIVOS CORPORATIVOS

1.1 Antecedentes

El manejo de los archivos e información dentro de la empresa xyz se encontraba en un total caos. Los usuarios que requerían una información especifica y se encontraba alojada en diferentes maquinas sin poseer siquiera una conexión de Red, tenían un serio inconveniente al momento de requerirla. La información se distribuía a los distintos equipos a través de medios de almacenamiento externos, sean estos: floppys, cd's, pendrives y Discos duros externos, lo cual originaba una gran perdida de tiempo e incomodidad entre los usuarios.

Los recursos informáticos de la empresa no eran aprovechados en su totalidad, dando como resultado una grave deficiencia en la productividad de la misma.

Los equipos no contaban con el mas mínimo nivel de seguridad, ya que cualquier usuario podía acceder como administrador e incluso manipular la información mas relevante para la empresa.

1.2 Problemática

Al no contar con una conexión de red, los usuarios no podían mantener una comunicación directa con equipos remotos de la empresa en los cuales se encontraba alojada la información solicitada, esto originaba pérdida de tiempo, pérdida de archivos de gran importancia para la empresa, dificultad y confusión al sacar un respaldo de la información e incluso falta de eficiencia en el cumplimiento de sus labores.

Falta de automatización en los procesos de la comunicación, debido al escaso aprovechamiento de los recursos informáticos existentes en la empresa.

Total libertad en los accesos a los equipos de la empresa, sin control alguno de cuentas de usuarios y permitiendo que cualquier empleado que tuviere frente de él, un computador pueda visualizar, copiar archivos sin control, e incluso modificar la información contenida en ese equipo.

Existían duplicación de archivos que no se encontraban actualizados, se daba el caso, de que un usuario utilizaba un archivo anterior que ya había

sido actualizado por otro usuario dando problemas de no tener la información real.

1.3 Solución

Instalar un servidor de archivos basado en la plataforma Linux por ser opensource genera un ahorro en pago de licencias tanto para el servidor como para los clientes.

Este servidor de archivo permite centralizar la información y poderla compartir entre los usuarios de la red, conservando la integridad de la misma logrando que esta se encuentre disponible en todo momento.

Estos archivos son almacenados en un centro de datos de alta seguridad y protegidos mediante una clave adicional que es proporcionada por el administrador del servidor de archivos.

Los usuarios podrán acceder a esta información para leerlos y modificarlos dependiendo tanto de los niveles de acceso y permisos otorgados por el administrador. Cada usuario tendrá una capacidad límite de almacenaje para su respectivo uso.

El Administrador podrá llevar un registro de usuarios al servidor es decir a través de logs estos registros serán almacenados a manera de bitácora.

El administrador podrá llevar un control de las cuotas de disco asignadas a cada usuario o grupo de usuarios.

1.4 Definición de un Servidor de Archivos

Un Servidor de Archivos proporciona una ubicación central en la red, en la que puede almacenar y compartir los archivos con usuarios de la misma red. Cuando los usuarios necesiten un archivo importante, como un plan de proyecto, podrán tener acceso al archivo que se encuentra en dicho servidor en lugar de tener que pasarlo entre distintos equipos, si los usuarios de la red necesitan tener acceso a los mismos archivos y aplicaciones accesibles a través de la red.

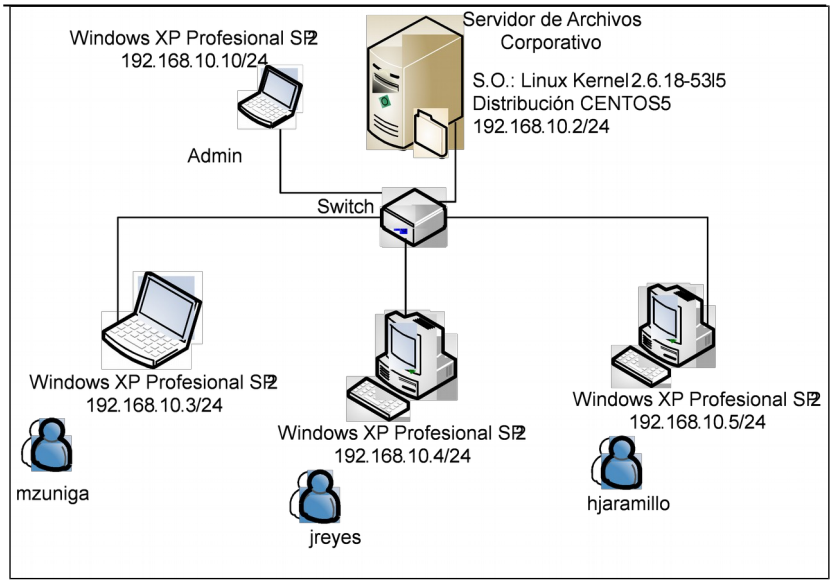


Figura 1.1 Arquitectura del servidor de archivos

1.5 Visión

Poder implementar nuestro sistema en las pequeñas y medianas empresas incluso en hogares que ya cuentan con una pequeña red, ofreciendo nuestro servicio con una solución que no requiere muchos gastos, para la gestión y control de la información de una manera segura y confiable.

1.6 Misión

Contribuir mediante una herramienta de administración al manejo y control de los archivos en una red, permitiendo al administrador la configuración de un

centro de datos de una manera amigable y dando la facilidad a cualquier usuario que pueda levantar su propio servidor de archivos.

1.7 Objetivos

1.7.1 Objetivo General

Implementación de un servidor de archivos que permita la administración y control de los mismos de manera eficiente, logrando así mantener toda la información centralizada, estableciendo las políticas de accesos y permisos correspondientes a los diferentes tipos de usuarios que manipularán la información requerida de acuerdo a su perfil accediendo desde cualquier maquina bajo una interfaz Web.

1.7.2 Objetivos Específicos

- Centralizar la información, lo cual permitirá una fácil y mejor administración de la misma.
- Brindar un entorno seguro de usuarios a nuestro servidor, logrando que la integridad de la información se mantenga siempre.

- Manejo de archivos de una manera eficiente y confiable entre los usuarios autenticados a través de una red de datos privadas.
- Proporcionar la seguridad en la transmisión de los datos entre los diferentes puntos.
- Permitir a los usuarios compartir la información necesaria para el cumplimiento de sus labores.
- Brindar la confianza necesaria a nuestros usuarios para el correcto acceso a la información requerida por los mismos.
- Acceder a la información en todo momento y lugar desde cualquier punto de la intranet o red privada.
- Permitir al administrador la creación un medio compartido para manipular correctamente dicha información.
- Tener la información al alcance de todos los usuarios de nuestra organización basándose en niveles de acceso.
- Centralizar la información y darle categorías a las mismas mediante una jerarquía empresarial.

1.8 Alcances.

- Instalación y configuración del servidor Linux para la creación y uso de un servidor de archivos.
- Instalación y configuración del paquete de servicios Samba para establecer un enlace seguro entre diferentes sistemas de archivos y lograr compartir información entre dos plataformas.
- Creación de una interfaz Web amigable desarrollada en tecnología java, que permita la fácil Administración del Servidor de Archivos Corporativos, ya sea al momento de definir usuarios, perfiles, contraseñas y demás atributos. Dicha aplicación tendrá dos entornos:
    - El entorno para el cliente el cual permitirá ver los recursos compartidos y,
    - El entorno para el administrador el cual permitirá la gestión de archivos, el ingreso de nuevos usuarios, gestión de accesos, gráficas estadísticas de consumo de quotas.
- Permitir la configuración y administración de recursos al igual que los permisos a usuarios.

- Implementación de servidor de dominio con autenticación integrando samba con openLDAP.

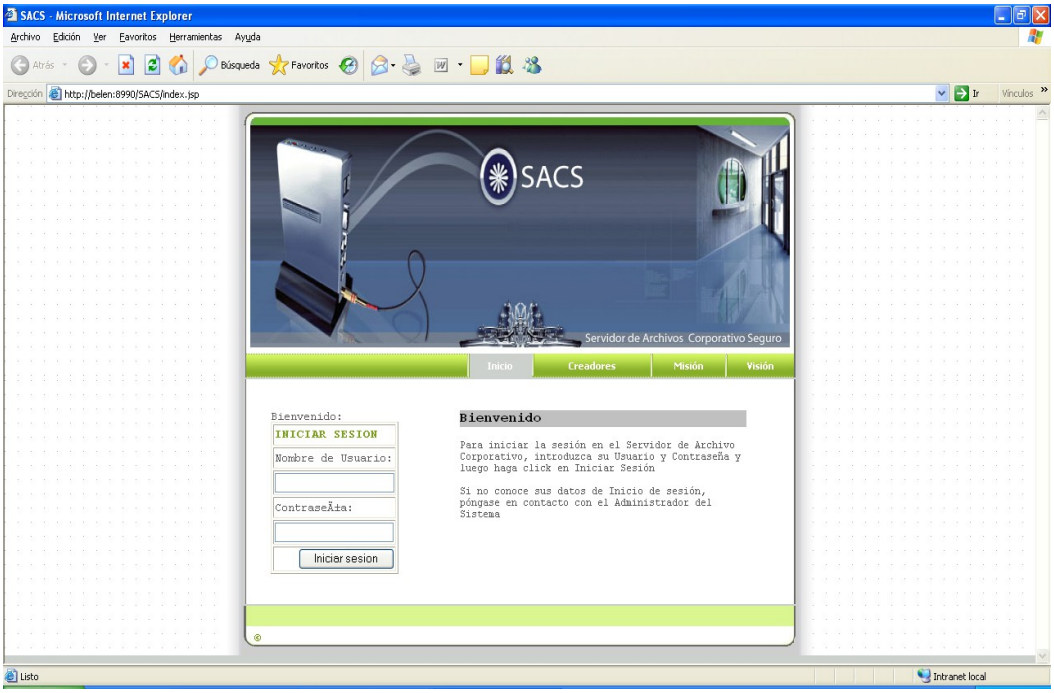


Figura 1.2 Interfaz Web de la aplicación

- Con respecto a las seguridades de manera interna se plantearán las reglas de Firewall (Iptables) adecuadas para evitar ataques a nuestro Servidor mediante el uso de bloqueos a través de IP's.
- Transmisión de archivos de manera segura con utilización de protocolos HTTPS con plataforma Windows.
- El usuario podrá activar el antivirus para realizar un scaneo completo de su directorio personal.
- El usuario la primera vez que se autentique la aplicación le exige que realice el cambio de la clave asignada por el administrador.

- Cada usuario tendrá un espacio limitado en disco o quota de usuario, para que pueda disponer del mismo.

- Se enviarán alarmas por correo electrónico a los usuarios cuando el sistema considere que su cuota de usuario esté por llenarse a su límite establecido. La aplicación controlara que al faltar el 5% de la cuota asignada, presentara un aviso al momento que el usuario ingrese a la aplicación y a si mismo enviara un correo avisándole que ha excedido el limite de cuota de disco y deberá eliminar archivos que el considere innecesarios.

- El administrador podrá sacar respaldo de las configuraciones, de todos los usuarios, respaldo de la base de datos utilizando la aplicación a traves de una conexión FTP.

- El administrador podrá visualizar de manera grafica los reportes de cuota, el espacio libre y el utilizado tanto de los usuarios como de los grupos.

1.9 Identificación de la alternativa de solución

Para la implementación y desarrollo del proyecto (Servidores de Archivos Corporativos) emplearemos herramientas como: Java JSP o Java Script para

la creación de las páginas Web interactivas. Estas herramientas son gratuitas, esto nos permitirá disponer de estos recursos sin incurrir en gastos adicionales.

Implementaremos un pequeño laboratorio donde instalaremos nuestra estructura tecnológica utilizando ordenadores propios de cada integrante del grupo. En estos ordenadores instalaremos nuestra plataforma de desarrollo, levantaremos los servicios necesarios que requiere nuestro proyecto bajo la plataforma Linux.

1.10 Recursos

1.10.1 Recursos de Hardware.

Emplearemos los siguientes recursos:

- 1 Computador (Servidor) (ver características en tabla 1.1)
- 2 Computadores (Clientes) (ver características en tabla 1.1)
- Equipos de red.
    - 1 Switch 10/100 Mbps
    - 3 Conexiones de red (Patch Cord)
    - 3 Tarjetas de red PCI 10/100 Mbps

1.10.2 Recursos de Software.

1.10.2.1 Recursos Software Servidor

- Herramienta de desarrollo Jdvestudio
- S.O Linux Distribución CentOs 5.0
- Paquete SAMBA 3.0.25b-0.el5.4.
- Paquete SAMBA-CLIENT 3.0.25B-0.el5.4.
- Paquete SAMBA-COMMON 3.0.25B-0.el5.4.
- Paquete openLDAP

1.10.2.2 Recursos Software Clientes

- S.O Windows XP Professional
- Microsoft Office

Características	Servidor	Clientes
Procesador	Intel Pentium IV 2.8Ghz	Intel Pentium IV 2.0Ghz
Mainboard	BioStar	MSI 845
Memoria RAM	512MB DDR 333Mhz	512MB DDR 333Mhz
Disco Duro	80Gb	80Gb
Amplitud del bus	32bits	32bits
Frecuencia bus	533Mhz	400Mhz
Monitor	SVGA 15”	SVGA 15”

Tabla 1.1 Características de Computadores

1.10.3 Recursos Humanos.

- Jorge Reyes Tomalá.
- Mónica Zúñiga Tapia
- Hernán Jaramillo Jiménez.

1.11 Metodología

Emplearemos la metodología de desarrollo de Software Orientado a Objetos utilizando el Modelo en Espiral con arquitectura n capas que abarca las siguientes etapas del ciclo de desarrollo del software.

1. Levantamiento de información e investigación preliminar.
2. Análisis.
3. Diseño
4. Desarrollo, Implementación y Pruebas del Sistema
5. Documentación y Manuales
6. Presentación

1.12 Cronograma de Actividades

1. Investigación Preliminar

    1.1 Consultas Bibliográficas

1.2 Consultas Por Internet

1.3 Consulta a profesionales

1.4 Ámbito de Estudio

1.5 Recursos de Hardware y Software

1.6 Revisión de Documentos Existentes

1.7 Identificación de los Objetivos

1.8 Identificación de los Alcances

2. Análisis

2.1 Levantamiento de la información

2.2 Entrevista

2.3 Investigación

2.4 Análisis de los Requerimientos

2.5 Análisis de los Recursos Humanos

2.6 Análisis de los requerimientos de Hardware y Software

2.7 Estudio de Factibilidad

2.8 Análisis Costo-Beneficio

2.9 Diccionario de Datos

2.10 Configuración del Servicio Samba

2.11 Elaboración de Casos de Uso

2.12 Elaboración de Diagramas de Flujo

Primera Presentación Avance

3 Diseño

3.1 Elaboración del Diseño de Datos

3.2 Elaboración del Diseño Arquitectónico

3.3 Diseño de la Interfaz
3.4 Elaboración del Prototipo

Primera Sustentación

4. Desarrollo Pruebas

4.1 Desarrollo

Segunda Sustentación

5.Implementacion.

6. Sustentación Final

CAPITULO 2
2 ANALISIS 17

2.1 Levantamiento de la Información 18

2.1.1 Entrevista 18

2.1.2 Investigación 18

2.2 Análisis de los requerimientos 19

2.2.1 Análisis de Recursos Humanos 19

2.2.2 Análisis de Recursos de Hardware y Software 19

2.3 Estudio de Factibilidad 20

2.3.1 Análisis Beneficio-Costo 20

2.3.1.1 Factibilidad Técnica 20

2.3.1.2 Factibilidad Operacional 21

2.3.1.3 Factibilidad Económica 22

2.4 Diccionario de Datos 22-23

2.5 Acerca de Samba 24

2.6 Configuración del Servicio Samba 25

2.7 Arquitectura Samba 23

2.8 Caso de Uso 25-28

2.9 Diagrama de Flujo 29-38

CAPITULO 2

2. Análisis

Esta etapa comprende una parte importante en del desarrollo del proyecto basado en el AOO (Análisis Orientado a Objetos) aquí es donde podremos tener una forma concreta de representar el conocimiento de los requisitos.

Dentro de este capitulo comenzaremos con una descripción de los casos de uso describiendo los escenarios y actores y la forma de cómo estos interactúan con la aplicación a desarrollar. El modelo de clases nos ayuda a trasladar la información de los casos de uso a una representación de clases.

Encontraremos también diagramas de flujo que permitirán poder tener una idea más clara de los distintos procesos que realizara la aplicación.

2.1. Levantamiento de la información

El Levantamiento de la información se llevo a cabo utilizando los recursos necesarios como la entrevista con personal y de investigación.

2.1.1 Entrevista

Para la utilización de este método de levantamiento de información previamente se llevaba una conversación para poder separar una cita con el especialista.

2.1.2 Investigación

En el método de investigación para el levantamiento de la información se consulto en la Web varias páginas relacionadas con las herramientas utilizadas para llevar a cabo este proyecto. Algunas páginas consultadas fueron www.linuxparatodos.com, www.samba.org, www.adictosaltrabajo.com, www.javasun.com

Se investigó también en revista como todo Linux, personalcomputer&Internet. Todo programación, etc. que nos ayudaron para adquirir conocimientos sobre algunos conceptos que desconocíamos referente a Linux, necesarios para el desarrollo del proyecto.

2.2. Análisis de los requerimientos

2.2.1 Análisis de recursos Humanos.

Las personas que van a tener acceso a nuestro Servidor de Archivos Corporativos, serán usuarios con sus respectivos permisos y usuarios Administradores con conocimientos básicos de Administración en Redes y conocimientos en Sistemas Operativos Linux.

2.2.2 Análisis de recursos de hardware y software

Para implementar el Servidor de Archivos Corporativos en una empresa dependiendo de la magnitud de la misma se considerara en el caso de una empresa pequeña que no es necesario adquirir un servidor de alto costo, podríamos levantar este servidor en un equipo con las siguientes características.

Procesador	Intel Pentium IV
Memoria RAM	512 MB
Disco Duro	120 GB
Monitor	SVGA 17''
DVDROM	
Tarjeta Red	10/100 Mbps

2.3. Estudio de la Factibilidad

El estudio de Factibilidad nos permitirá tener una visión del porcentaje del grado de factibilidad que tenemos para que el desarrollo de nuestro proyecto se lleve a cabo.

La hemos clasificado en tres sub-niveles para obtener un detalle específico de estos factores.

2.3.1. Análisis Costo-Beneficio

Para llevar a cabo este análisis Costo-Beneficio se realizó subgrupos para obtener el costo o beneficio que cada uno de estos subgrupos refleja.

2.3.1.1. Factibilidad Técnica

Consiste en analizar si existe la estructura técnica necesaria para llevar a cabo el proyecto y el costo que esto representaría.

En cuanto a la tecnología necesaria para realizar este proyecto, los computadores para el Servidor, Clientes y Equipos para la red local eran de propiedad de los desarrolladores del proyecto por lo que no hubo que realizar gastos en cuanto a los equipos (Ver tabla 1 y 2)

EQUIPOS	Costo
Computadores	$ 0.00
Switch 10/100	$ 0.00
Patch Cord	$ 0.00
Impresora	$ 0.00
TOTAL	**$ 0.00**

Tabla 2.1 Gastos de Recursos de Hardware

Software	Costo
S.O Linux Distribución CentOS 5	$ 0.00
IDE NetBeans 6.0	$ 0.00
Apache-tomcat-6.0.14.tar	$ 0.00
Samba	$ 0.00
Samba-Client	$ 0.00
Samba-Common	$ 0.00
TOTAL	**$ 0.00**

Tabla 2.2 Gastos de Recursos de Software

2.3.1.2. Factibilidad Operacional

Es factible llevar a cabo el proyecto, debido a que los gastos operacionales que son necesarios no se encuentran en un alto grado.

La tabla 2.3 muestra algunos de los gastos operacionales realizados o necesarios.

Gastos Operacionales	Costo
Movilización de equipos.	$ 30.00
Movilización del personal	$ 540.00
Energía Eléctrica	$ 120.00
Teléfono	$ 60.00
Internet	$ 60.00
TOTAL	**$ 800.00**

Tabla 2.3 Gastos de Recursos de Software

2.3.1.3. Factibilidad Económica

Aquí analizamos si contamos con los recursos económicos necesarios para la realización del proyecto. En base a los resultados de la factibilidad técnica y factibilidad operacional

Podemos observar que en cuanto a gastos por recursos técnicos tenemos un total de $0.00 dólares. Los únicos gastos que incurrirán son los netamente relacionados con la implementación del lugar de trabajo y gastos de manutención del mismo.

Estos gastos se encuentran en un nivel de poco riesgos como para ser un obstáculo en la realización del proyecto. Esto resultaría de la inversión de cada uno de los miembros desarrolladores del proyecto.

2.4 Diccionario de Datos.

Nombre	Smb.conf
Descripción	Archivo de Configuración de Samba

Nombre	Workgroups
Descripción	Nombre del grupo al que pertenece una red

Nombre	Netbios

Descripción	Nombre que le asigna al equipo o PC de una red

Nombre	Host Allow
Descripción	Política que indica las PC que tienen acceso

Nombre	smbpasswd
Descripción	Archivo para creación añadir usuarios con contraseña
Donde Se Usa	Configuración y creación de contraseña

Nombre	Users
Descripción	Usuarios de nuestra red

Nombre	Computer
Descripción	Clientes de nuestra red

Nombre	Smb-client
Descripción	Diversos clientes para el protocolo SMB

Nombre	Smb-commom
Descripción	Ficheros necesarios para cliente y servidor
Donde Se Usa	Configuración de samba

2.5 Acerca de Samba

Antes de explicar la configuración de este servicio indicaremos, que el Servicio Samba es un conjunto de programas originalmente creados por Andrew Tridgell y actualmente obtenidos por The SAMBA Team**,** bajo Licencia Pública General GNU y que implementan el protocolo SMB en sistemas basados sobre Unix**.** Además, sirve como reemplazo total para Windows NT, NFS, o Servidores NetWare.

SMB (acrónimo Server Message Block), es un protocolo, del nivel de Presentación del modelo OSI de TCP/IP, creado en 1985 por IBM. Algunas veces es referido también como CIFS (acrónimo de Common Internet File System), tras ser renombrado por Microsoft en 1998.

2.6 Configuración del Servicio Samba

Para la mayoría de los casos la configuración de Samba como Servidor de Archivos es suficiente ya que fue creado con un objetivo: ser un respaldo definitivo para Windows como servidor en una red local. Esto por supuesto, requiere principalmente del fichero smb.conf y de algunos procedimientos adicionales dependiendo de las necesidades de la red local.

2.7 Arquitectura Samba

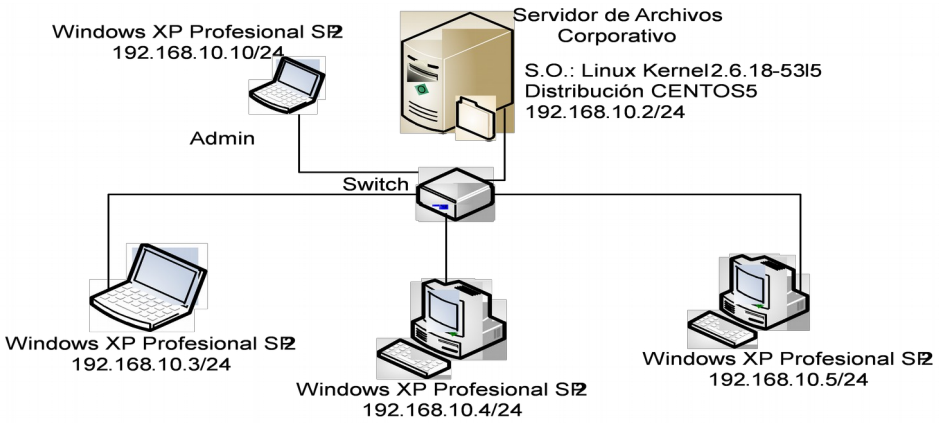


Figura 2.1 Arquitectura Samba

2.8 Casos de Usos

Durante la etapa del análisis los casos de usos nos ayudaron a identificar los escenarios y actores que están involucrados al momento de interactuar con la aplicación web para administrar el servidor de archivos corporativo. En nuestro proyecto se identificaron dos actores principales que en forma generalizada serian los usuarios pero dependiendo de los permisos que se le otorga a estos usuarios podríamos determinar que un actor es el Administrador y otro serian los usuarios de la red.

La figura 2.2 muestra los 2 escenarios principales de nuestra aplicación bajo un actor de manera generalizada que en este caso son todos los usuarios.

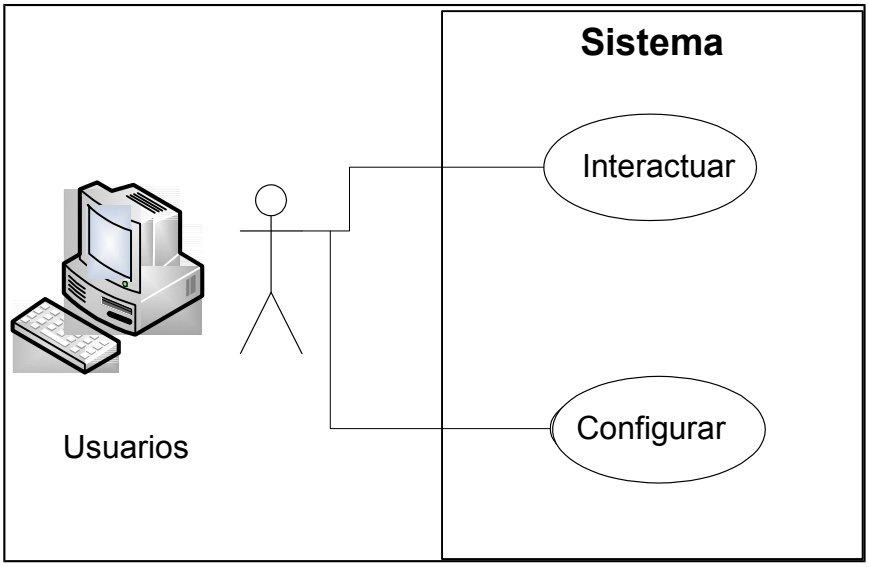


Figura 2.2 Caso de Uso de forma General

La figura 2.3 mostramos gráficamente el escenario que esta definido para los Usuarios de la Red que no tengan perfil de Administradores o que no pertenezcan al grupo de Administradores. Este escenario muestra la forma de los usuarios al interactuar con la aplicación y determina las limitaciones que estos tienen.

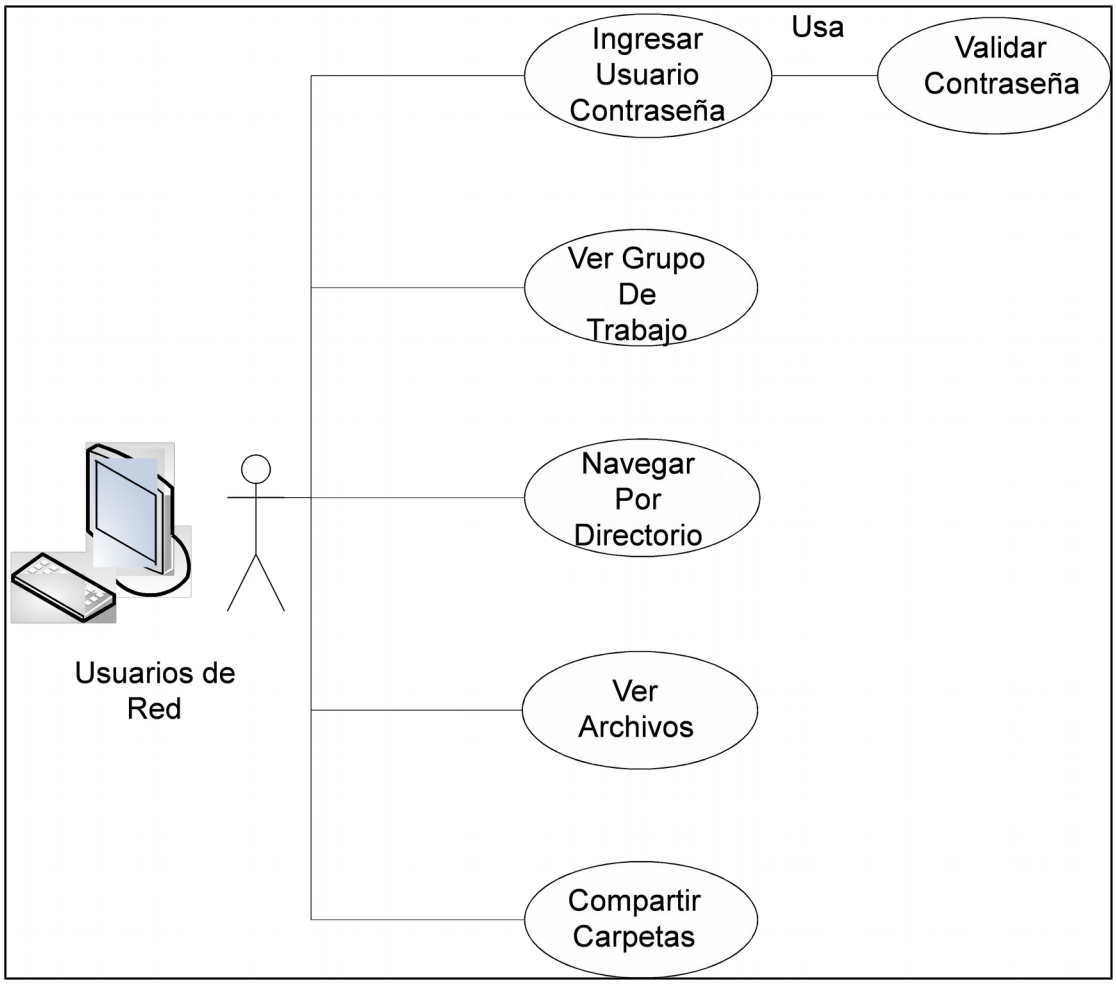


Figura 2.3 Escenario de Usuarios de Red

Otro Actor identificado son el Administrador o los Administradores usuarios de Red pero que pertenecen al grupo de administradores y tienen permisos para todos los procesos de la aplicación. La figura 2.4 muestra gráficamente el escenario y los procesos que el administrador puede acceder.

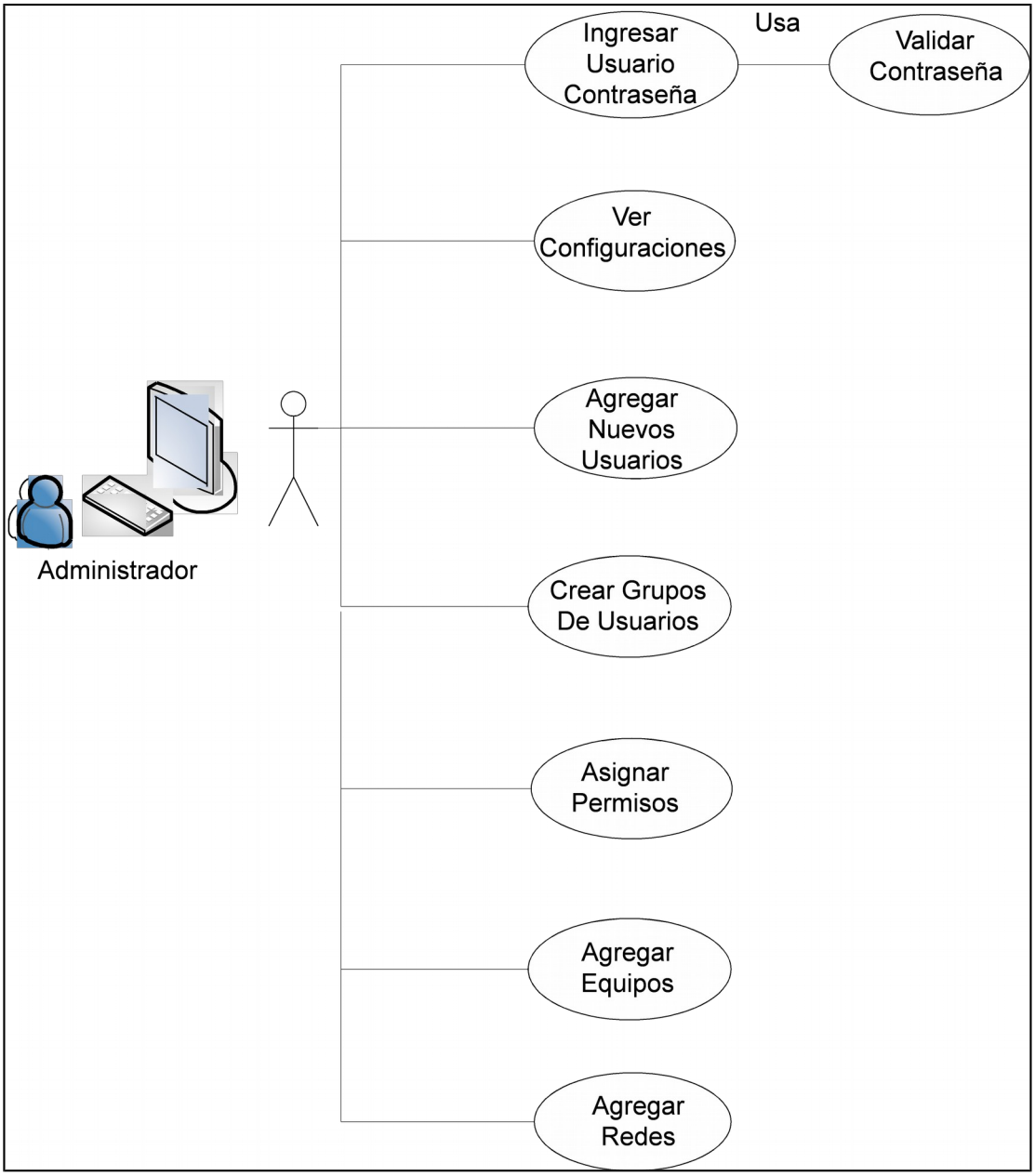
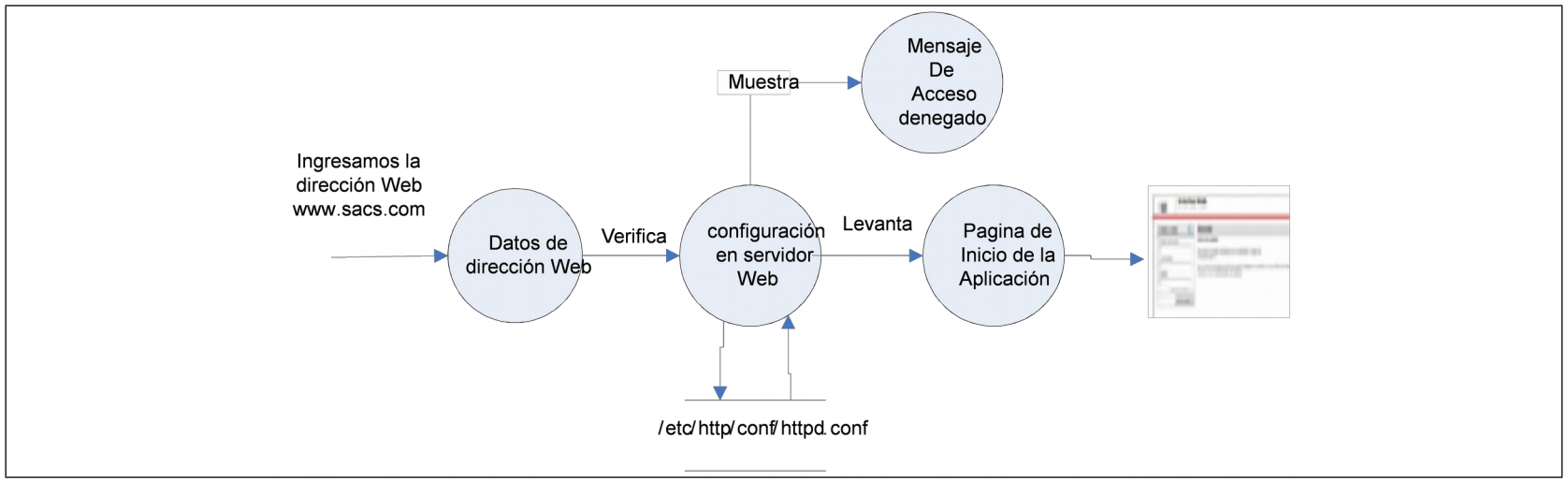


Figura 2.4 escenario del usuario Administrador

2.9 Diagrama de Flujo.
Conectividad de Equipo



Conectividad de equipos.

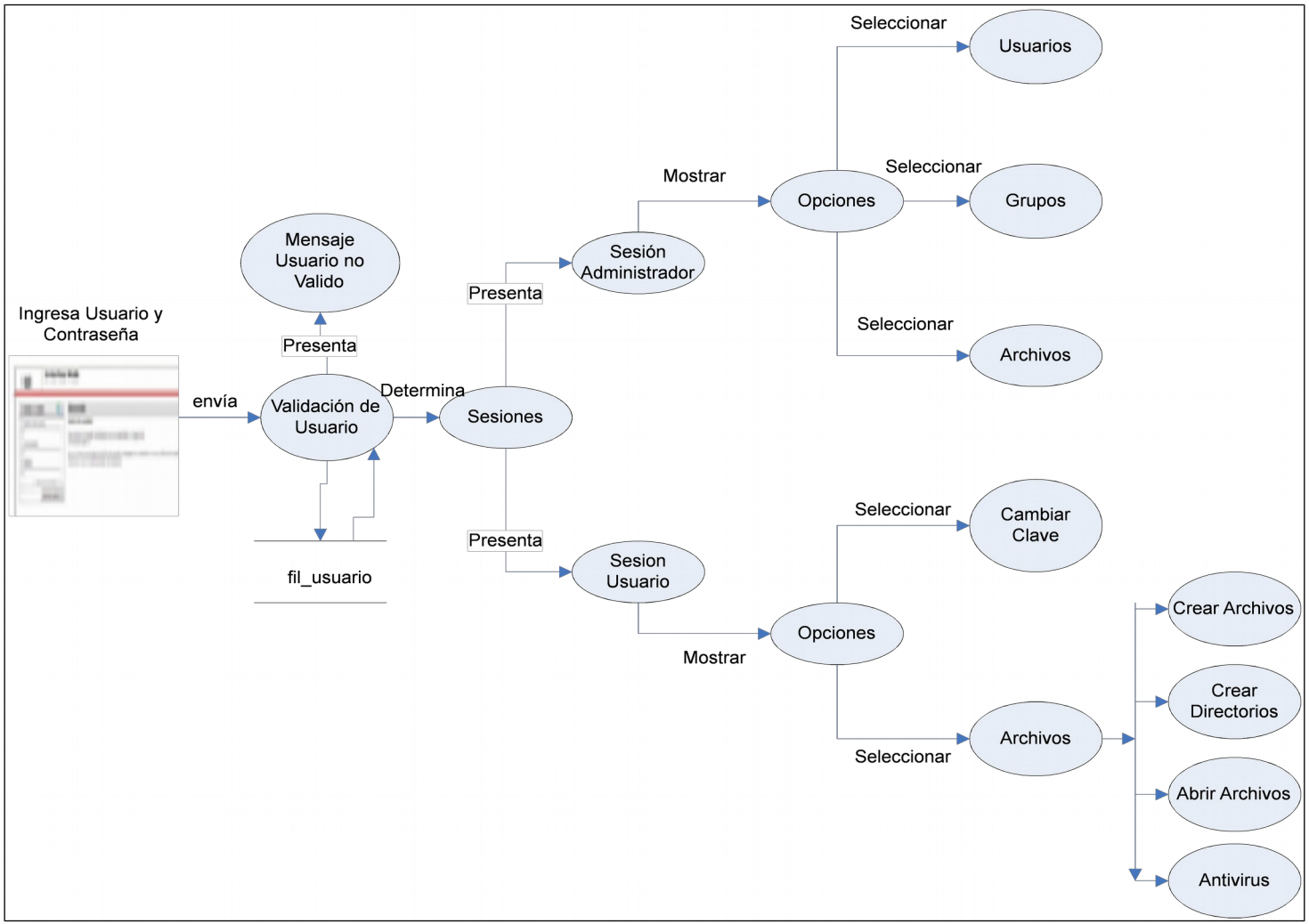
Este diagrama muestra los procesos iniciales que son transparente para el usuario son procesos que se realizan al momento de abrir un navegador web desde una maquina cliente y digitar la direccion web del servidor http://www.sacs.com.

Si el servidor web se encuentra levantado y la configuracion del mismo es la correcta se mostrara la pagina de inicio de la aplicación la cual se encuentra configurado en el archivo /etc/http/conf.d/http.conf.

Notas:

- Dentro del servidor web en el archivo de configuracion del httpd.conf se crearon los sitios virtuales con la direccion www.sacs.com.
- Tambien se determino un sitio virtual para el acceso a la pagina con protocolo seguro https://www.sacs.com/SACS
- Caso contrario se mostrara un error de pagina no encontrada o de seevidor.

Modulo de Inicio.

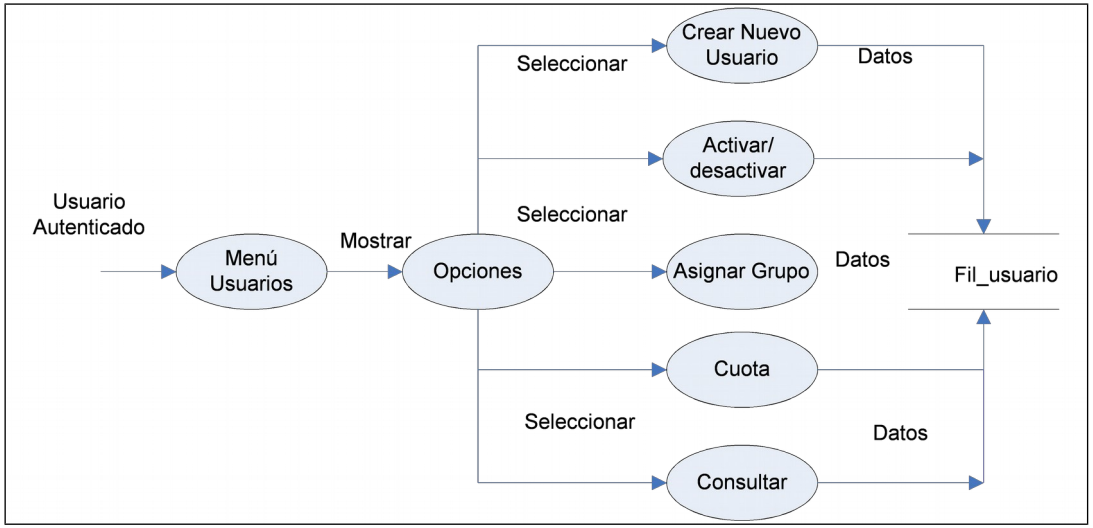


Modulo de Inicio de Sesion.

Describe los procesos de validación de los usuarios a través de un nombre y un password el cual previamente debe existir o estar dado de alta en el servidor. Estos procesos permiten realizar la coneccion a la base de datos cuyos usuarios que estan creados en el servidor estan guardados en la tabla fil_usuario si se encuentra el usuario en la tabla se abre la sesion de usuario o de administrador según el perfil del usuario autenticado.

- La Aplicación permitirá un nivel de seguridad ante cualquier persona que desee ingresar a nuestro servidor.
- El usuario podrá ingresar desde cualquier equipo de la red.
- Una vez activas la sesión el administrador o usuario podrán hacer uso de los menús de opción que tiene la aplicación

Modulo Usuario.



Modulo Usuario.

Describe los procesos de Mantenimiento de los usuarios.

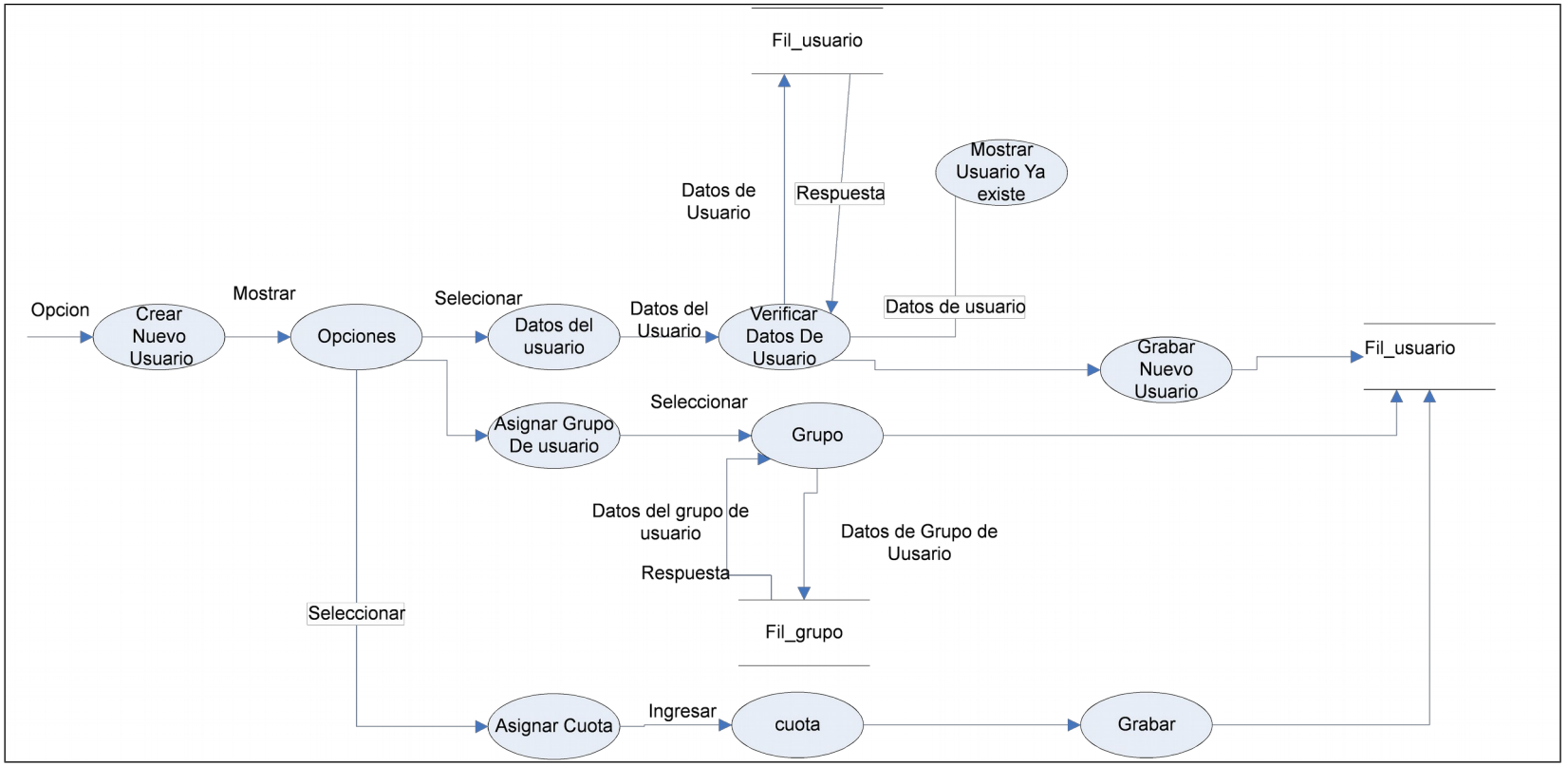
En este modulo el Administrador podrá crear un nuevo usuario, editar un usuario existente, o activar/desactivar un usuario.

Cada proceso que se encuentran en este modulo a su vez tiene subprocesos que se realizan durante la implementación y uso del modulo.

Notas:

- Un usuario de la red no podrá acceder a este modulo si no tiene perfil de administrador o pertenece al grupo de Administradores.
- Al desactivar un usuario este no podra autenticarse al dominio y no podra utilizar la aplicación.

Creación de Usuarios



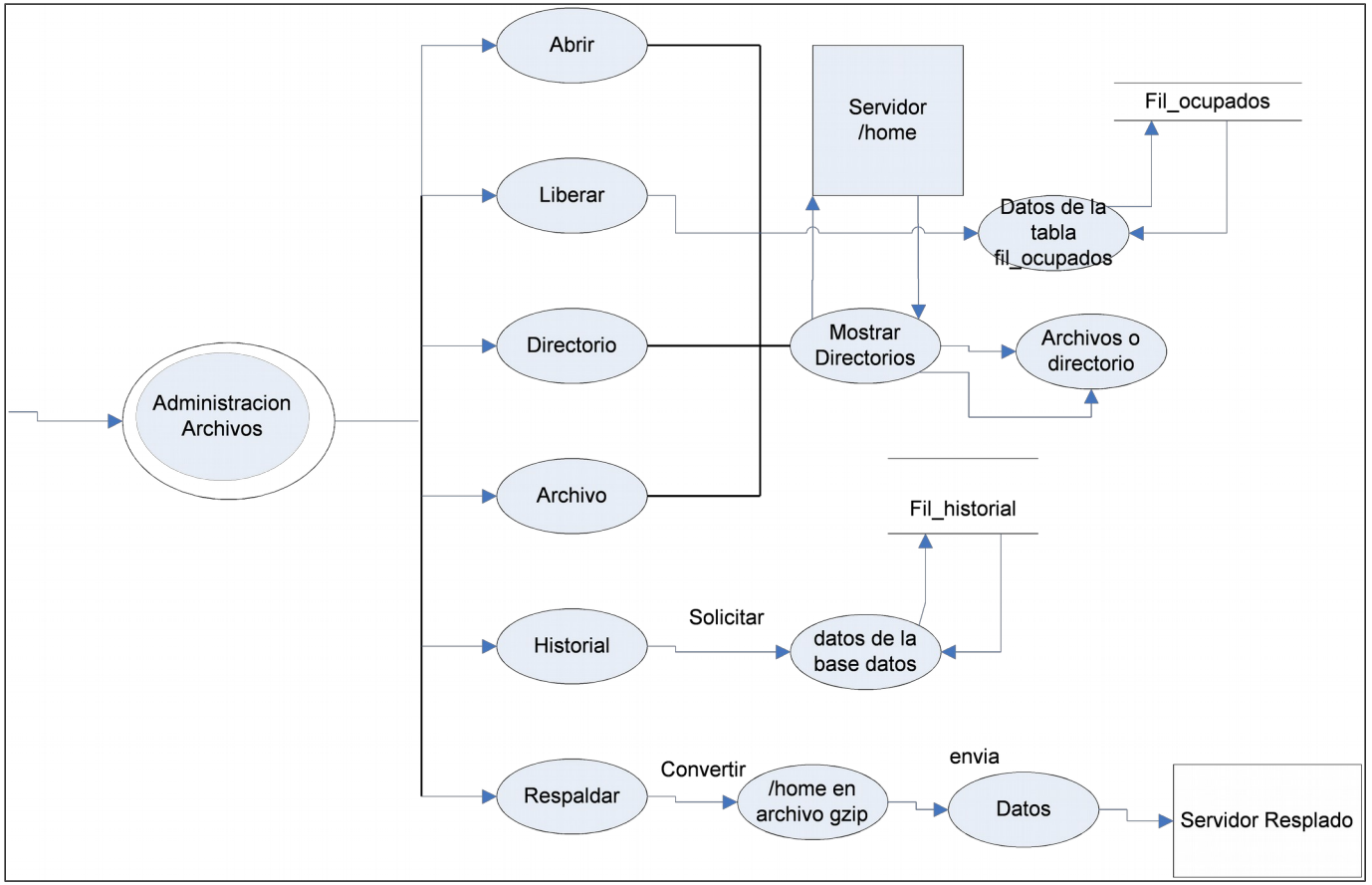
Creación de Usuario

Detalla los subprocesos que se realizan al seleccionar la creación de un nuevo usuario. Se ingresan datos como el nombre, contraseña y este proceso verifica si el usuario ya existe en la red, en caso de existir presenta un mensaje de que existe y sale del proceso, luego se selecciona el grupo de usuario al que va pertenecerse no existe el grupo se crea en ese momento. Al final se asigna la cuota de disco al usuario creado.

Notas:

- El Nombre del usuario la primera letra debe ser escrita con mayúscula, no se debe utilizar caracteres especiales.
- La contraseña debe contener al menos 11 caracteres pueden ser de tipo alfanumérica, (a-z, A-Z, 0-9, - @ &, etc.).
- Los grupos de Usuario deben tener nombre del grupo o departamento al que pertenece dentro de la corporación ej. Ventas, Gerentes, Financiero, etc.

Administración de Recursos Compartidos



Administración Recursos Compartidos

Detalla los procesos que se realizan al administrar los recursos
La administración del recurso realiza tres proceso principales que son crear un nuevo directorio o archivo modificar este directorio o poder visualizar el contenido del directorio o archivo.

Al abrir un archivo desde un cliente utilizando la aplicación web, el equipo cliente determina la aplicación necesaria para poder abrir estos archivos como los .doc .xls .pdf.

Al momento de crear un archivo elegimos en que carpeta dentro del home del usuario deseo crear el archivo o directorio. En los archivos debemos especificar la extension del archivo, si el nombre del archivo existe este se creara como ocupado y no podremos abrirlo para editarlo.

Al crear un archivo este inicialmente se crea en el servidor pero vacio si editamos el archivo este subira toda la informacion que ingresamos al archivo en el momento de liberarlo en ese momento ya se creara completo en el servidor.

DISEÑO **40**

3.1 Diseño de Datos **41**

3.1.1 Estructura de Datos **42**

3.1.1.1 Tablas **42**

3.1.1.2 Claves Primarias **43**

3.1.1.3 Claves Foreanas **43**

3.2 Diseño Arquitectónico **43**

3.3 Diseño de Interfaz **44**

3.3.1 Interfaz Principal **45-58**

.

CAPITULO 3

3. Diseño

El diseño comienza con el modelo de los requerimientos, luego se trabaja por transformar este modelo y obtener cuatro niveles de detalles de diseño: la estructura de Datos, la arquitectura del sistema, la representación de la interfaz y los detalles a nivel de componentes.

3.1 Diseño de Datos.

El Diseño de Datos nos permite trasformar el modelo de información creado en el análisis, en las estructuras de datos que se van a requerir para implementar nuestra aplicación.

La figura 3.1 muestra los objetos involucrados y detectados en el análisis los cuales en esta etapa de diseño se creara una base de datos para la administración de usuarios.

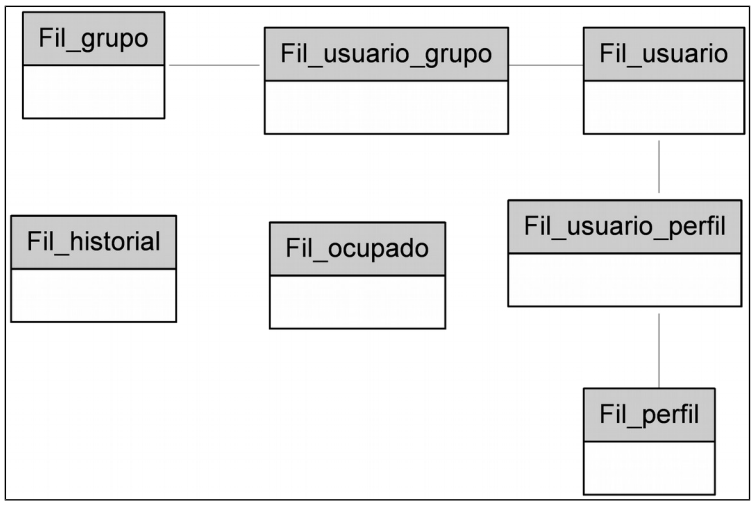


Figura 3.1 Diseño de Datos

3.1.1 Estructura de Datos.

3.1.1.1 Tablas.

fil_grupo		
id_Grupo	Int(4)	Primari Key
descripción	Varchar(20)	NULL
estado	Varchar(1)	NULL
cuota	double(2,4)	NULL

fil_usuario		
Id_usuario	Varchar(15)	Primari Key
nombre	Varchar(15)	NULL
clave	Varchar(15)	NULL
estado	Varchar(1)	NULL

fil_perfil		
id_perfil	Varchar(4)	Primari Key
descripción	Varchar(20)	NULL
estado	Varchar(1)	NULL

fil_usuario_perfil		
id_usuario	Varchar(15)	Primari Key

Id_perfil	Int(4)	NULL

fil_usuario_grupo		
id_usuario	Varchar(15)	Primari Key
id_grupo	int(4)	Primari Key
cuota	Double(4,2)	NULL

fil_historial		
archivo	Varchar(50)	NULL
id_usuario	Varchar(15)	Primari Key
direccion	Varchar(200)	NULL

fil_ocupado		
archivo	Varchar(50)	NULL
id_usuario	Varchar(15)	Primari Key
direccion	Varchar(200)	NULL

3.1.1.2 Claves Primarias

fil_grupo

Primari Key (Id_Grupo)

fil_usuario

Primari Key (id_usuarios)

fil_perfil

Primari Key (id_perfil)

3.1.1.3 Claves Foráneas

fil_usuario_grupo

Foreing Key(Id_Grupo) References (fil_grupo)

Foreing Key(Id_Grupo) References (fil_usuario)

3.2 Diseño Arquitectónico.

En esta fase del diseño se realiza luego de una vez definido el diseño de datos basado en esto se procede a la derivación de una o mas representaciones de la estructura arquitectónica de la aplicaron.

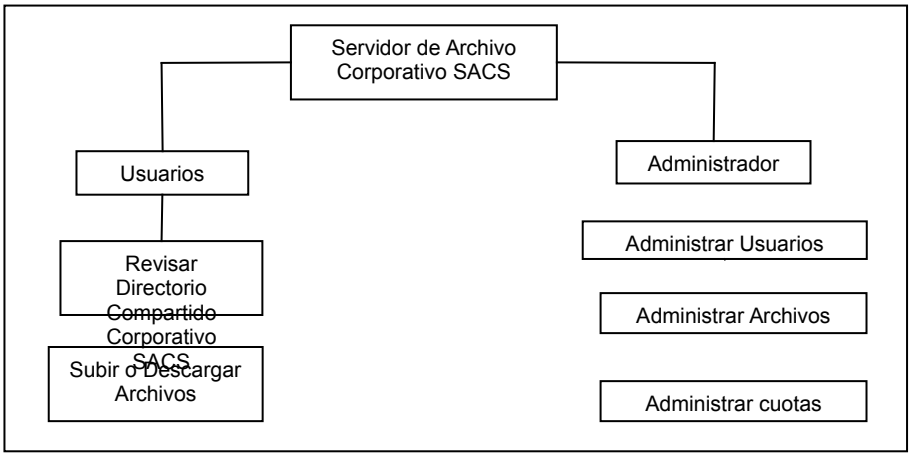


Figura 3.2 Diseño Arquitectónico

3.3 Diseño de Interfaz

3.3.1 Interfaz Principal.

Interfaz principal nos permite la comunicación de los usuarios con el servidor. Debemos digitar el nombre del usuario y la contraseña luego damos Clic en el botón iniciar sesión y en este momento enviamos nuestros datos a través de la red para que sean verificados por el servidor y este permita establecer una comunicación y así poder abrir nuestra sesión sea de administrador o de un usuario de red.

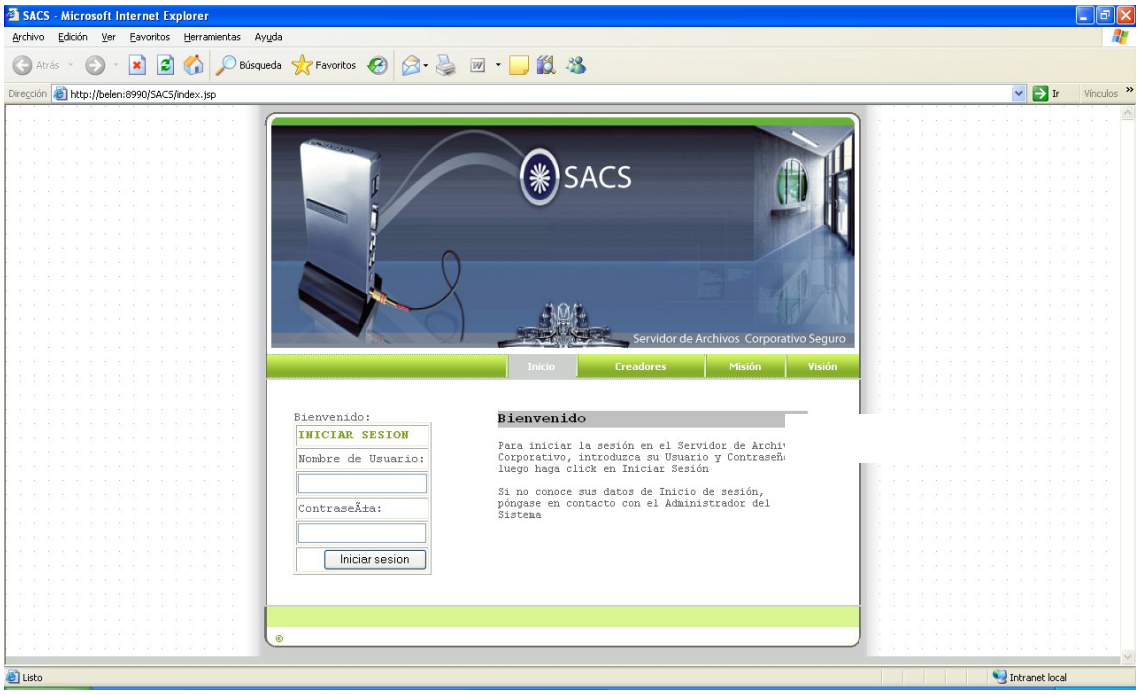


Figura 3.2 Diseño de la Interfaz

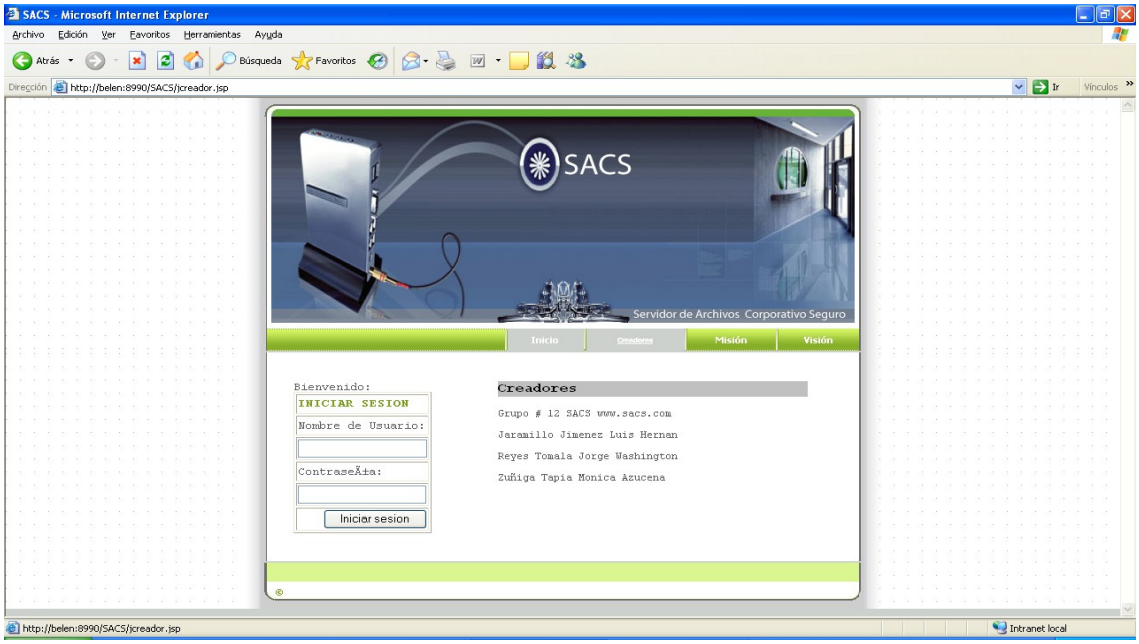


Figura 3.3 Creadores

Luego, el usuario podrá visualizar la Misión del Proyecto

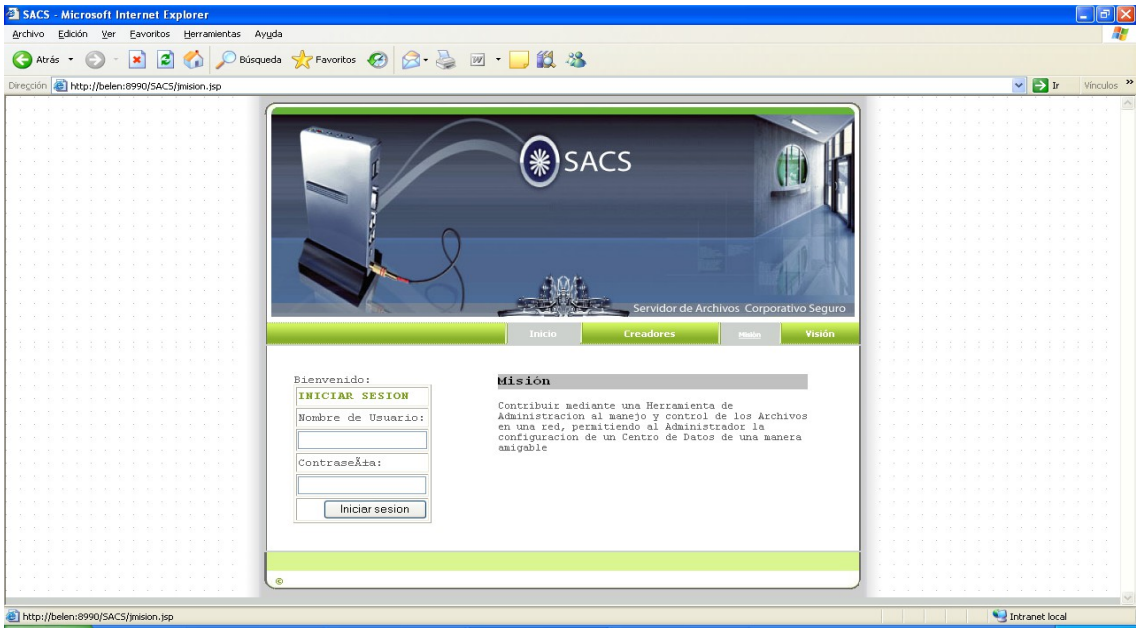


Figura 3.4 Misión

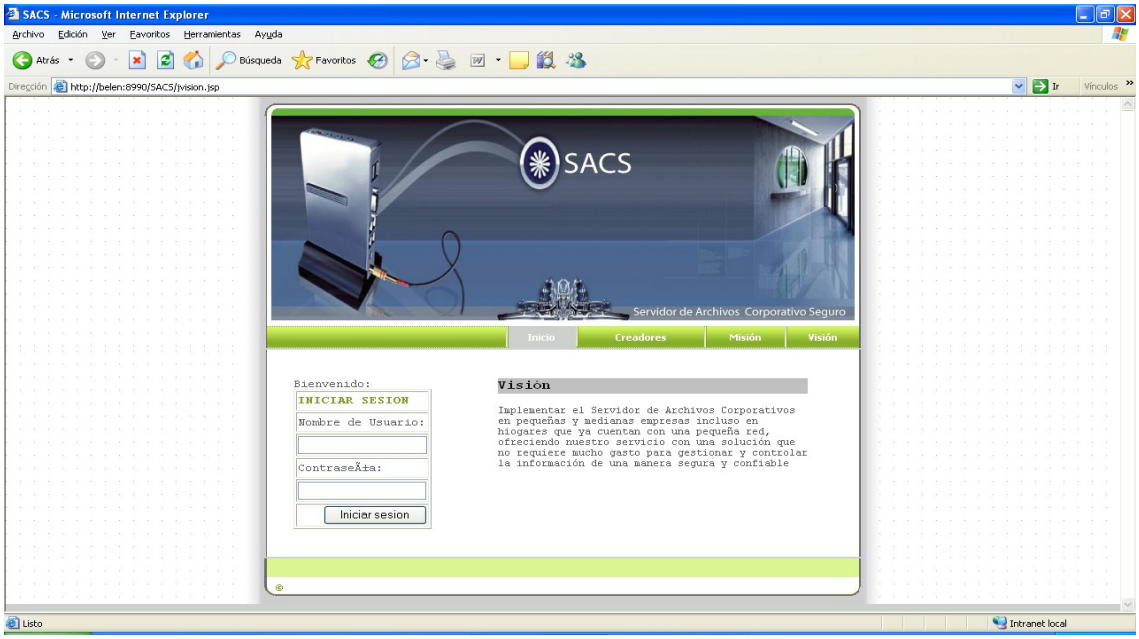

SACS - Microsoft Internet Explorer
Archivo Edición Ver Favoritos Herramientas Ayuda
Atrás Búsqueda Favoritos
Dirección http://belen:8990/SACS/jvision.jsp
Ir Vínculos
SACS
Servidor de Archivos Corporativo Seguro
Inicio
Creadores
Misión
Visión
Bienvenido:
INICIAR SESION
Nombre de Usuario:
ContraseÃ±a:
Iniciar sesion
Visión
Implementar el Servidor de Archivos Corporativos en pequeñas y medianas empresas incluso en hiogares que ya cuentan con una pequeña red, ofreciendo nuestro servicio con una solución que no requiere mucho gasto para gestionar y controlar la información de una manera segura y confiable
Listo
Intranet local

Figura 3.5 Vision

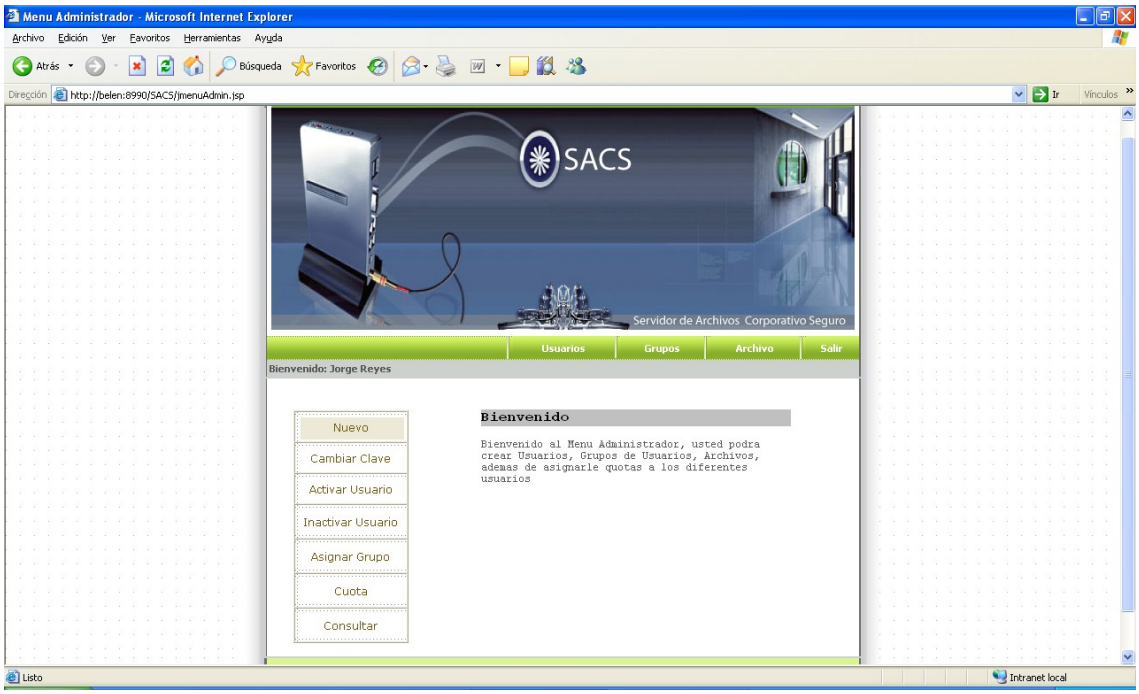


Figura 3.6 Menú Administrador

.

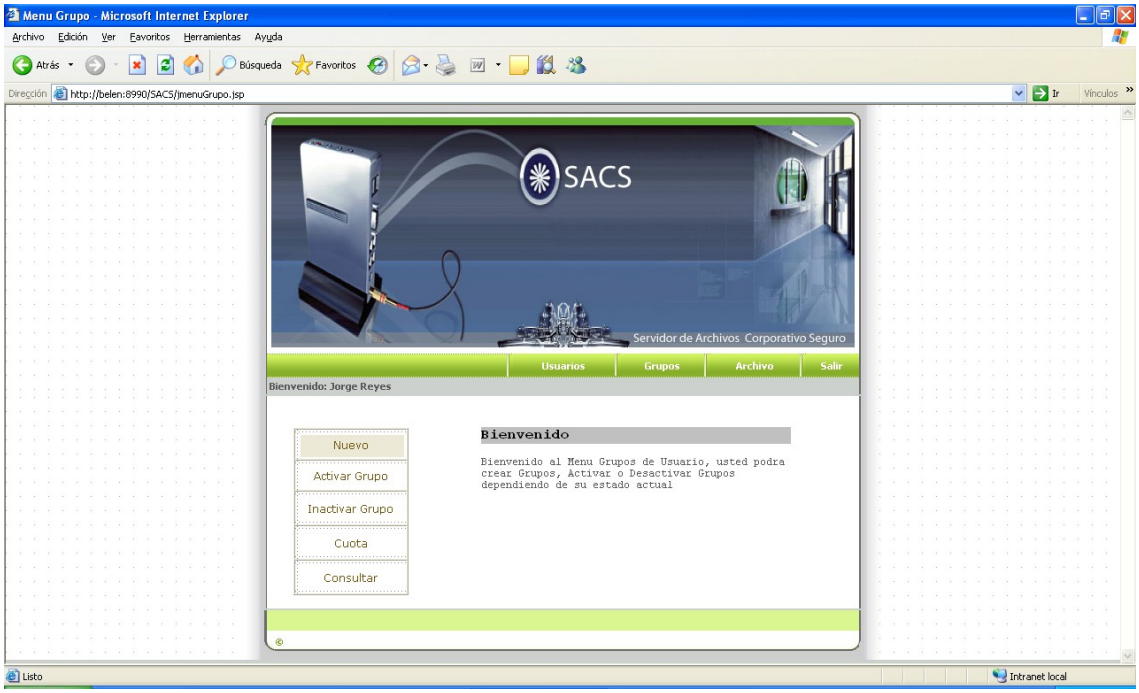


Figura 3.7 Grupos

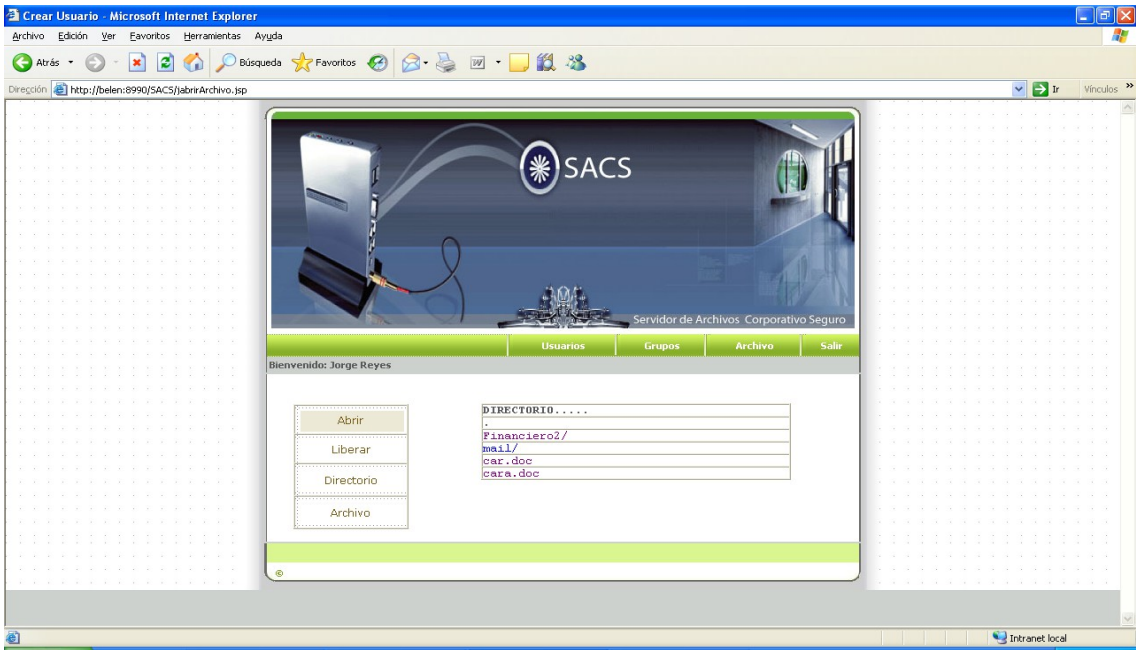


Figura 3.8 Menú Archivos

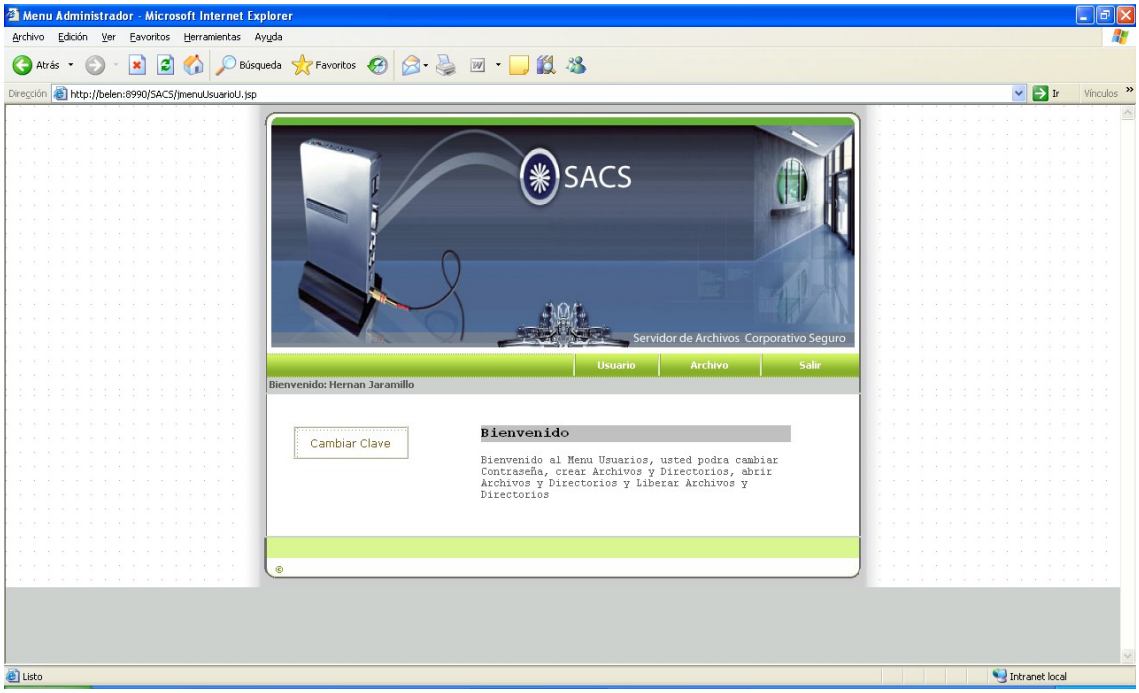


Figura 3.9 Menú Usuario

CAPITULO 4

DESARROLLO Y PRUEBAS DEL SISTEMA 50

4.1 Configuración del servidor de Archivos Corporativo 51

4.1.1 Instalación, Configuración del DNS 51

4.1.1.1 Pruebas del servidor DNS 52

4.1.2 Instalación, Configuración de samba como PDC 52-54

4.1.3 Instalación, Configuración OpenLDAP 54-58

4.1.4 Integración Samba con OpenLDAP 58-60

4.1.5 Instalación, Configuración de SMBLDAP-TOOLS 60-63

4.1.6 Instalación, Configuración servidor VSFTP 63-64

4.1.7 Asignación de Cuotas 64

4.1.8 Configuración servidor de correo sendmail 65-67

4.1.9 configuración servidor Web HTTP 67-69

4.2 Desarrollo e implementación Base Datos 69

4.2.1Instalacion MySQL 69

4.2.2 Creación de la base de Datos 70-72

4.2.2.1 Conexión a la base de Datos 73

4.3 Desarrollo de Aplicación Web Para la Administración del proyecto. 74-75

4.3.1 Clases. 75

4.3.2 Scripts 76-79

CAPITULO 4

4 DESARROLLO Y PRUEBAS DEL SISTEMA

El desarrollo de nuestro proyecto se compone principalmente de 3 etapas para su realización las cuales veremos a lo largo de este capitulo. Estas etapas son las siguientes:

1. Configuración del Servidor de Archivos Corporativos
   dns, samba, openLdap, web, vsftp, http, iptables.
2. Creación de la base de datos.

mysql

3. Desarrollo de la interfaz web para la administración del proyecto.

   Jdk, oc4j, jdevstudio10132

4.1 Configuración del servidor de Archivos Corporativos

4.1.1 Configuración del servidor nombre de dominio DNS.

Para la instalación del servidor de nombres de dominios se deben instalar los siguientes paquetes:

bind-9.3.3-7.el5.i386.rpm

bind-libs-9.3.3-7.el5.i386.rpm

bind-chroot-9.3.3-7.el5.i386.rpm

bind-utils-9.3.3-7.el5.i386.rpm

caching-nameserver-9.3.3-7.el5.i386.rpm.

Una vez instalados los paquetes creamos los archivos de zonas.

Para nuestro dominio sacs.com creamos el archivo de zona de reenvió **/var/named/chroot/var/named/sacs.com.zone, (**ver Anexo 2.1.1 sacs.com.zone).

La zona inversa para la red 192.168.15.0/24 **/var/named/chroot/var/named/15.168.192.in-addr.arpa.zone**. (ver Anexo 2.1.2 15.168.192.in.addr.arpa.zone)

Luego editamos los parámetros de configuración en el archivo **/etc/named.conf.(**ver Anexo 2.1.3 named.conf)

Por ultimo realizamos el arranque de los demonios del servidor DNS y realizamos pruebas de su funcionamiento.

Para dar inicio a los demonios del servidor utilizamos los siguientes comandos:

#service named start // inicia demonio del servidor DNS

#service named stop // detiene demonio del servidor DNS

#service named restart // reinicia demonio del servidor DNS

Si queremos que el servidor se inicie al momento de arranque del sistema operativo se debe ejecutar el siguiente comando.

#chkconfig named on .

4.1.1.1 Pruebas del servidor DNS.

Dentro de una terminal ejecutamos el siguiente comando.

nslookup muestra el nombre del servidor sacs.com y nos permite ingresar nombres de maquinas de nuestra red o si no las direcciones ip. Para que el servidor resuelva los nombres.

Debemos configurar el archivo /etc/resolv.conf

Search sacs.com

Nameserver 192.168.15.254

4.1.2 Instalación y configuración del servidor samba como PDC.

Paquetes necesarios:

samba-3.0.23c-2.i386.rpm. //Servidor SMB

//Clientes para el servidor.

samba-client-3.0.23c-2.i386.rpm SMB

//Archivos Comunes part servidor SMB

samba-common-3.0.23c-2.i386.rpm.

Una vez instalados los paquetes necesarios abrimos el archivo de configuración con el comando vi /etc/samba/lmhosts donde añadimos el nombre del servidor junto con la dirección ip del mismo e incluso podríamos poner las demás maquinas conectadas al servidor.

192.168.15.254 sacs

Posteriormente abrimos el archivo /etc/samba/smb.conf y configuramos los parámetros necesarios para poder levantar el servidor samba.

workgroup = sacs //nombre del dominio

server string =Servidor Archivo // descripcion del servidor

netbios name = server // nombre del servidor

Security = user

local master = yes //servidor examinador del dominio

os level = 65

domain master = yes //define al servidor maestro del dominio

preferred master = yes //servidor maestro preferido

domain logon = yes //servidor permite a estaciones autentica r contra samba

logon script = logon.cmd // guion que tendrá las estaciones al conectarse con samba.

(ver Anexo 2.2.1 smb.conf)

4.1.3 Instalación y configuración de openLdap como servidor de autenticación.

El servidor openLdap nos permitirá crear contenedores de información para tenerlas almacenadas y poderlas administrar de una manera mas eficiente y segura.

Lo primero es definir la estructura o árbol directorio Ldap. Comenzamos con las bases para nuestro árbol:

dc=sacs, dc=com

El “dc” representa componente de dominio esto lleva un orden en nuestro proyecto la dirección web termina en .com por eso la base de nuestro árbol es dc=com seguido del nombre del dominio dc=sasc. Luego formamos las unidades organizacionales “ou” en el proyecto manejamos 3 unidades organizacionales ou=Users, ou=Groups, ou=Computers nuestro árbol Ldap quedaria:

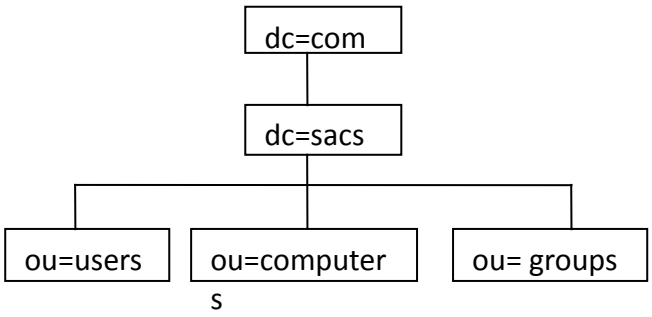


Figura 4.1 Arbol Ldap

Para implementar un servidor openldap debemos instalar los siguientes paquetes:

openldap-2.3.27-5.i386.rpm

openldap-clients-2.3.27-5.i386.rpm

openldap-servers-2.3.27-5.i386.rpm

authconfig-5.3.12-2.el5.i386.rpm

authconfig-gtk-5.3.12-2.el5.i386.rpm

utilizaremos el comando rpm -hiv nombre_del_paquete.rpm

para la instalación de los paquetes.

Después de instalados los paquetes debemos seguir una serie de procedimientos mostrados a continuación.

Debemos crear una carpeta y darle permisos de acceso al usuario y grupo Ldap.

mkdir /var/lib/ldap/autenticar

chmod 700 /var/lib/ldap/autenticar

chown ldap.ldap /var/lib/ldap/autenticar.

Creamos la clave que se le asignara al usuario administrador del directorio con el siguiente comando.

slappasswd y generara una clave encriptada, la cual será utilizada cuando editemos los parámetros de configuración en el archivo /etc/openldap/sladp.conf

{SSHA}b8JF2tZ+ST+w5tWM+s42MafnJaUMaXQN.

Editamos el fichero /etc/openldap/sladp.conf (ver Anexo 2.3.1 sladp.conf).

Luego iniciamos el demonio del servidor openLdap con el comando

#service ldap start.

#chkconfig ldap on // inicia al arranque del sistema operativo.

Editamos el fichero

/usr/share/openldap/migration/migrate_common.ph

modificamos los valores de las variable

$DEFAULT_MAIL_DOMAIN, $DEFAULT_BASE

(Ver Anexo 2.3.2 migrate_common.ph).

Debemos crear el objeto base.ldif que tendrá el resto de los datos, del directorio.

/usr/share/openldap/migration/migrate_base.pl > base.ldif

Insertamos información en el directorio

ldapadd -x -W -D 'cn=Administrador, dc=sacs, dc=com' -h 192.168.15.254 -f base.ldif

Debemos crear también los ficheros para almacenar la información de los grupos, de las cuentas incluyendo claves de acceso.

/usr/share/openldap/migration/migrate_group.pl /etc/group group.ldif

/usr/share/openldap/migration/migrate_passwd.pl /etc/passwd passwd.ldif

Ingresamos información a estos archivos.

ldapadd -x -W -D 'cn=Administrador, dc=sacs, dc=com' -h 192.168.15.254 -f group.ldif

ldapadd -x -W -D 'cn=Administrador, dc=sacs, dc=com' -h 192.168.15.254 -f passwd.ldif.

por ultimo debemos configurar que la autenticación se la realice utilizando openLdap utilizando una herramienta como el authconfig-gtk.

Digitamos en una terminal authconfig-gtk damos enter y saldrá la ventana de configuración.

Figura 4.2 configuración de la autenticación.

4.1.4 Integración Samba con openLdap.

En este punto debemos realizar unos cambios en los archivos de configuración tanto en el archivo /etc/openldap/sladp.conf como en el archivo /etc/samba/smb.conf.

(ver Anexo 2.2.1 y Anexo 2.3.1).

Debemos copiar las schemas de samba en el directorio de schemas de openLdap.

cp /usr/share/doc/samba-*/LDAP/samba.schema

/etc/openldap/schema/

Modificamos el archivo sladp.conf agregamos unas cuantas lineas.

Include /etc/openldap/schema/samba.schema

Index sambaSID eq

index sambaPrimaryGroupSID eq

index sambaDomainName eq

index default sub

También debemos configurar el archivo de clientes NSS /etc/ldap.conf (ver Anexo 2.3.2 ldap.conf cliente (NSS)).

Archivo cliente Ldap /etc/openLdap/ldap.conf (ver Anexo2.3.3 ldap.conf cliente (LDAP).

Se realizan cambios en el archivo smb.conf de samba (ver Anexo 2.2.1 smb.conf).

Ahora debemos hacerle saber a samba el password del administrador del openLdap con el siguiente comando.

smbpasswd -w password.

Luego realizamos un test a samba con el comando testparm y reiniciamos samba.

4.1.5 Instalación y configuración de smbldap-tools

Smbldap-tools es una herramienta que nos permite administrar la integración de samba con openLDAP

Para la instalación del servicio smbldap-tools necesitamos tener instalados los siguientes paquetes:

instalamos el RPM de http://dag.wieers.com/

release/rpmforge-release-0.3.6-1.el5.rf.i386.rpm

rpm -Uvh rpmforge-release-0.3.6-1.el5.rf.i386.rpm

luego editamos el fichero /etc/yum.repos.d/rpmforge.repo :

[rpmforge]

name = Red Hat Enterprise $releasever - RPMforge.net - dag

#baseurl = http://apt.sw.be/redhat/el5/en/$basearch/dag

mirrorlist = http://apt.sw.be/redhat/el5/en/mirrors-rpmforge

#mirrorlist = file:///etc/yum.repos.d/mirrors-rpmforge

enabled = 0

protect = 0

gpgkey = file:///etc/pki/rpm-gpg/RPM-GPG-KEY-rpmforge-dag

gpgcheck = 1

Dejamos deshabilitado el repositorio por defecto (enabled=0).

Luego de esto, debemos instalar paquetes adicionales con sus respectivas dependencias

perl-Compress-Zlib.i386 0:1.42-1.fc6.

perl-Convert-ASN1.noarch 0:0.20-1.1

perl-Crypt-SmbHash.noarch 0:0.12-1.2.el5.rf

perl-Digest-SHA1.i386 0:2.11-1.2.1

perl-HTML-Parser.i386 0:3.55-1.fc6

perl-HTML-Tagset.noarch 0:3.10-2.1.1

perl-IO-Socket-SSL.noarch 0:1.01-1.fc6

perl-Net-SSLeay.i386 0:1.30-4.fc6

perl-URI.noarch 0:1.35-3

perl-XML-NamespaceSupport.noarch 0:1.09-1.2.1

perl-XML-SAX.noarch 0:0.14-5

perl-ldap.noarch 0:0.33-1.el5.rf

perl-libwww-perl.noarch 0:5.805-1.1.1

smbldap-tools.noarch 0:0.9.1-1.2.el5.rf

Debemos conocer SID de nuestro servidor Samba con el comando: net getlocalsid

SID for domain LINUX is: S-1-5-21-XXXXXXXXXX-XXXXXXXXXX el cual lo utilizaremos mas adelante en la configuracion.

Procedemos a configurar el acceso smbldap-tools a la BD de openLDAP en /etc/smbldap-tools/smbldap_bind.conf: (ver anexo 2.4.1 smbldap_bind.conf)

configuración principal en /etc/smbldap-tools/smbldap.conf : (ver anexo 2.4.2 smbldap.conf).

Finalmente debemos crear la estructura del directorio LDAP

Utilizamos el siguiente comando

smbldap-populate -a Administrator

En este caso no se llamara root por defecto, sino que se le cambio al parámetro -a Administrator.

Listo ya podemos utilizar los comandos del smbldap-tools de la siguiente manera:

smbldap-useradd -a -m -F "" usuariowin
smbldap-passwd usuariowin

Reiniciamos samba y openLdap:

service ldap restart
service smb restart

4.1.6 Instalación y configuración de servidor VSFTP

Vsftp servidor FTP seguro utilizado para la transferencia de archivos de manera segura utilizado para implementar en servidores de archivos.

Podemos controlar el ancho de banda, la utilización del directorio exclusivo para el usuario y es mas sencillo de configurar.

Comprobamos si tenemos instalado el paquete con:

#rpm -q vsftp

Verificamos los ficheros de configuración en las siguientes rutas.

/etc/vsftpd.user_list Lista que definirá usuarios a enjaular o no a enjaular, dependiendo (ver Anexo 2.5.1 vsftpd.user_list)

/etc/vsftpd/vsftpd.conf Fichero de configuración. (Ver Anexo vsftpd.conf).

Para ponerlo operativo debemos iniciar el demonio o servicio vsftpd.

#service vsftpd start // iniciamos servidor ftp

#chkconfig vsftpd on // inicie al arrancar el sistema operativo.

4.1.7 Asignación de Cuotas de disco.

La utilización de cuotas de disco permite administrar eficientemente el espacio compartido en disco por múltiples usuarios. Las cuotas restringen la capacidad

de los usuarios para acceder hacia los recursos de sistema, tales como bloques (asignación de unidades). Cuando una cuota es excedida se aplica una política determinada por el administrador. Las

cuotas se administran por sistema de archivos individuales y son únicas para usuarios o grupos.

Debemos agregar al fichero /etc/fstab los parámetros usrquota, grpquota a la partición donde esta el directorio /home.

LABEL=/home /home ext3 defaults,usrquota,grpquota 1 2

Luego remontamos para que tenga efecto los cambios con el comando:

#mount -o remount /home

Debemos crear los ficheros para almacenar la información y el estado de las cuotas en cada partición

cd /home touch aquota.user aquota.group quota.user quota.group.

4.1.8 Configuración de servidor de correo.-

Utilizaremos un agente de transporte de correo como lo es el sendmail el más popular, utilizado por los servidores del mundo.

Para implementar el servidor de correo es necesario instalar los siguientes paquetes:

sendmail-8.13.8-2.el5.i386.rpm

sendmail-cf-8.13.8-2.el5.i386.rpm

dovecot-1.0-1.2.rc15.el5.i386.rpm //servidor de POP3 e IMAP basados sobre Unix™ y está diseñado con la seguridad como principal objetivo.

m4-1.4.5-3.el5.1.i386.rpm

make-3.81-1.1.i386.rpm

Luego editamos el fichero **/etc/mail/local-host-names** en el cual debemos poner todos los alias que tendrá el servidor que estamos configurando.

mail.sacs.com

servidor.mail.sacs.com.

Luego editamos el archivo de configuración /etc/mail/sendmail.mc

(Ver Anexo 2.6.1 sendmail.mc

Descomentamos la siguiente línea para filtrar los spam

dnl FEATURE(‘accept_unresolvable_domains’)dnl.

Debemos establecer la mascara que utilizara todo el correo que emitamos desde nuestro servidor. Añadiremos la siguiente línea.

MASQUERADE_AS(`sacs.com')dnl

Definimos los dominios a los cuales se esta permitido enviar correo

para esto debemos generar el archivo /etc/mail/relay-domains.

sacs.com

mail.sacs.com

Abrimos el archivo **/etc/mail/access** y agregamos algunas líneas para definir quienes podrán hacer uso de nuestro servidor de correo para poder enviar mensajes **(ver Anexo 2.6.2 /etc/mail/Access.)**

Por ultimo debemos convertir el fichero sendmail.mc en sendmmail.cf y compilar access para generar otro en formato de base de datos a fin de ser utilizado por Sendmail:

 # cd /etc/mail

 # make

Terminada la configuración hacemos que se reinicie el servicio o demonio del sendmail.

#service sendmail restart.

4.1.9 Instalación y configuración de servidor web HTTP.

Servidor web permite levantar nuestra aplicación web y ofrecerles a los clientes una interfaz amigable a través del protocolo HTTP.

Debemos tener instalado el siguiente paquete:

httpd-2.2.3-6.el5.centos.1.i386.rpm

Debemos configurar el archivo **/etc/httpd/conf/httpd.conf** (ver Anexo 2.7.1 /etc/httpd.conf).

Las directivas de este archivo controlan las operaciones de un Apache server, define sus parámetros, y configura los host virtuales.

Localizamos el parámetro serverAdmin e ingresamos el mail del administrador.

ServerAdmin root@localhost

En la opción serverName ingresamos la dirección ip del servidor y el puerto por el que se comunica el servidor.

ServerName www.sacs.com :80

en la opción serverRoot ingresamos la configuración del servidor errores y archivos de log.

ServerRoot "/etc/httpd"

Luego creamos un virtual hosts para poder presentar nuestra aplicación web www.sacs.com definimos parámetros como la ruta de la página seguido del puerto 8888, nombre del servidor etc.

```
<VirtualHost *:80>
    ServerAdmin root@sacs.com
    DocumentRoot /var/www/sacs
    ServerName www.sacs.com
    ProxyRequests Off
    ProxyPassReverse /SACS
http://www.sacs.com:8888/SACS
    ProxyPass /SACS http://www.sacs.com:8888/SACS
    ProxyPreserveHost On
    Redirect /index.jsp http://www.sacs.com/SACS
#    Redirect http://www.sacs.com
http://www.sacs.com
#    ErrorLog logs/dummy-host.example.com-error_log
#    CustomLog logs/dummy-host.example.com-
access_log common
</VirtualHost>
```

4.2 Desarrollo e implementación de la base de Datos.

Para la implementación de nuestro proyecto el desarrollo de la base de Datos se la realizo en MySQL por ser openSource. MySQL es el gestor de base de Datos multiusuario, multiplataforma, confiable, rápida y compacta.

4.2.1 instalación de MySQL

Como primer instancia se realizo la instalación del paquete mysql-5.0.22-2.1.i386.rpm y sus dependencias, para llevar a cabo esta instalación se utilizo el comando rpm -hiv mysql-5.0.22-2.1.i386.rpm.

Una vez instalada la base de datos debemos saber como arrancar el demonio de MySQL, detenerlo y poder alzarlo al momento de reiniciar el sistema operativo esto se lo realiza con los siguientes comandos:

#service mysqld start //inicia servicio de mysqlserver.

#service mysqld stop //detiene servicio de mysqlserver.

#service mysqld restart //reinicia servicio de mysqlserver

#chconfig mysqld on //inicia servicio al arrancar el sistema operativo.

La primera vez que se inicia el servidor de base de datos el usuario Root no tiene permisos para poder ingresar a la línea de comandos y poder realizar alguna acción por lo que se debe dar el permiso respectivo de la siguiente manera:

mysqladmin -u root password nueva-clave-de-acceso

Ahora ya podremos utilizar la línea de comandos para crear nuestro diseño de la base de datos

4.2.2 Creación de la base de Datos

La base de datos consta de las siguientes tablas:

fil_usuario

fil_usuario_grupo

fil_grupo

fil_perfil

fil_usuario_perfil

fil_historial

fil_ocupado

Tabla: Fil_Usuario

Maneja los datos relacionados con el usuario del sistema.

permite almacenar la cuota de disco asignada.

Tabla: fil_usuario_grupo

Permite tener al usuario referenciado con el índice del grupo al que pertenece.

Campos:

Id_usuario

Id_grupo

Id_cuota

Tabla: fil_grupo

Permite manejar los datos de los grupos del sistema.

Campos:

Id_grupo

Descripción

Estado

Cuota

Tabla: fil_perfil

Permite manejar los perfiles o rol de los usuarios del sistema en nuestro proyecto manejamos solo perfil Administrador y perfil de usuario común.

Campos:

Id_usuario

Id_perfil

Tabla: fil_historico

Esta tabla permite almacenar el nombre del archivo el id del usuario y la direccion del archivo para poder presentar como reporte de historico de archivos.

Tabla: fil_ocupado

Esta tabla permite guardar los archivos que se encuentran ocupados tambien nos permite poder controlar el uso de los archivos y poder realizar el proceso de liberacion de archivos.

```
package fileserver;
import java.sql.Connection;
import java.sql.SQLException;
public class Conexion{
    private Connection connection;
    private String idUsuarioSistema;
    public Conexion()throws SQLException
  {
      java.sql.DriverManager.registerDriver(new com.mysql.jdbc.Driver());
    }
    public Connection getConnection() throws SQLException {
        if(connection==null || connection.isClosed())
        {
 connection=java.sql.DriverManager.getConnection("jdbc:mysql://db:3308/fileserver", "fileserver",
"fileserver");
            connection.setAutoCommit(false);
        }
        return connection;
    }
public String getIdUsuarioSistema() {
        return idUsuarioSistema;
    }
    public void setIdUsuarioSistema(String idUsuarioSistema) {
        this.idUsuarioSistema = idUsuarioSistema;
    }
    public String commit()
    {
      String err=null;
      try
      {
        Connection c=this.getConnection();
        c.commit();
      }
      catch (Exception ex)
{
        err=ex.toString();
        ex.printStackTrace();
      {
        err=ex.toString();
        ex.printStackTrace();
      }
      return err;
    }
  public String rollback()
  {
    String err=null;
    try
    {
      Connection c=this.getConnection();
      c.rollback();
    }
  catch (Exception ex)
     {
      err=ex.toString();
      ex.printStackTrace();
    }    return err;
  }
}
```

4.3 Desarrollo de la Aplicación web para la administración del proyecto.

El desarrollo de la aplicación será a través de un Modelo de Espiral, tomando en cuenta los riesgos que se pueden presentar a lo largo del ciclo de vida del proyecto.

El Modelo en Espiral es el óptimo para el proyecto, ya que permite ir comprobando cada etapa del desarrollo que este cumpliéndose correctamente.

Es flexible en cuanto a cambios durante el desarrollo de las interfaces.

La metodología usada para el desarrollo de este sistema, es la Orientada a Objeto.

Para la realización de las GUI (Interfaces Gráficas de Usuarios) se utiliza el Modelo de Prototipos.

El desarrollo de la Aplicación web primero se desarrollo la estructura o arquitectura en la que nuestra aplicación iba a implementarse. La Aplicación web se desarrollo en base al lenguaje de programación java usando jsp, javascripts para la creación de las paginas por lo tanto que teníamos que tener instalado la estructura para comenzar el desarrollo de la aplicación. Con el comando **rpm -hiv jdk-6u5-linux-i586-rpm** instalamos el jdk de java que contiene las librerías y la maquina virtual para nuestro proyecto.

Adicionalmente utilizamos la herramienta de desarrollo jdevstudio10132 que incluye el jdeveloper, la instalación de esta herramienta es muy sencilla copiamos la carpeta jdevstudio10132 en /usr, esta herramienta trabaja con servidor web oc4j a través del puerto 8888. Dentro de la carpeta jdevstudio10132 existe el archivo jdeveloper.exe dando un click en este archivo se levanta el entorno de desarrollo para ejecutar la aplicación web debemos levantar el servidor oc4j dando un click en el archivo **start_oc4j**

ubicado en **/usr/jdevstudio10132/jdev/bin/start_oc4j** el cual se ejecutara en una terminal.

4.3.1 clases.

Usuario.

Grupo.

UsuarioPerfil.

Jftp.

Perfil.

4.3.2 Scripts

Scripts desarrollados para trabajar con comandos directos hacia nuestro servidor Linux.

- **Screarusuario.**

  Permite crear usuarios del dominio con su clave de acceso llamando al script Sclave

```
#!/bin/bash
#parametros donde:
#       $2 = Usuario
#       $3 = Clave
#        $1 = Grupo
smbldap-useradd -a -m -F "" $1
SClave $1 $2
```

- **SClave**

```
#!/bin/bash
#parametros donde:
#       $1 = Usuario
#       $2 = Clave
smbldap-passwd $1<< EOF
$2
$2
$2
EOF
```

- **SCrearGrupo**

permite crear grupos de mi dominio, a la vez llama al script que da los permisos a este grupo,

```
#!/bin/bash
#parametros donde:
#       $1 = Grupo
#       $2 =
smbldap-groupadd $1
mkdir /home/samba/$1
SCrearPermisos $1
SCrearGrpSamba $1
```

- **SCrearPermisos**

Damos permisos a las carpetas creadas dentro del home de cada usuario dentro del servidor.

```
#!/bin/bash
#parametros donde:
#       $1 = Grupo
chown -R root.$1 /home/samba/$1
chmod -R 2770 /home/samba/$1
```

- **SCrearGrupoSamba**

```
#!/bin/bash
#parametros donde:
#       $1 = Grupo
#       $2 =
```

```
cd /etc/samba
echo "["$1"]" >>smb.conf
echo "comment = Archivos comunes de "$1 >>smb.conf
echo "path = /home/samba/"$1>>smb.conf
echo "valid users = @"$1>>smb.conf
echo "write list = @"$1>>smb.conf
echo "force group = "$1>>smb.conf
echo "create mode = 770">>smb.conf
echo "directory mode = 770">>smb.conf

service smb restart
```

- ***SAsignarQuotaGrupo***

```
#!/bin/bash
#parametros donde:
#       $1 = Grupo
#       $2 = quota

#***************ASIGNAR CUOTA DE GRUPO******************
setquota -g $1 0 $2 0 0 /home
```

- **SDesabilitarUsuarios**

```
#!/bin/bash
#parametros donde:
#       $1 = Usuario
#       $2 = Grupo
smbldap-groupmod -m $1 $2
```

- **SDesabilitarGrupo.**

```
#!/bin/bash
#parametros donde:
#       $1 = Grupo
chmod -R 700 /home/samba/$1
```

CAPITULO 5

5 Implementación del Sistema 80

5.1 Introducción 81

5.2 Elementos Físicos 82

5.3 Elementos Lógicos 82

5.4 Elementos Humanos 82

5.5 Infraestructura 83

CAPITULO 5

5 Implementación del Sistema

Este capitulo nos describe los recursos utilizados para la implementación del proyecto.

5.1 Introducción

La implementación del Servidor de Archivos Corporativo se realizó en base un a Herramienta muy poderosa como es OpenSource

Para llevar a cabo la implementación de nuestro sistema se necesito de varios elementos, como citamos a continuación:

Elementos Físicos

Elementos Lógicos

Recurso Humano

Infraestructura

Capacitación a los Usuarios

5.2 Elementos Físicos

Entre los elementos físicos que se utilizo para la implementación de este sistema, tenemos los siguientes:

- 1 Computador (Servidor) (ver características en tabla 1.1)
- 2 Computadores (Clientes) (ver características en tabla 1.1)
- Equipos de red.
    - 1 Switch 10/100 Mbps
    - 3 Conexiones de red (Patch Cord)
    - 3 Tarjetas de red PCI 10/100 Mbps

5.3 Elementos Lógicos

En el servidor deberán estar instalados y correctamente configurados el Sistema Operativo CentOS 5.0, la Base de Datos Mysql y todos los servicios necesarios, como son:

Dns: Servidor de Nombres de Dominio

Samba: Protocolo que permite a los sistemas Windows compartir ficheros, archivos y directorios.

OpenLdap: es un protocolo para consulta y modificación de servicios de directorio que se desempeñan sobre TCP/IP.

Vsftp: es un equipamiento lógico utilizado para implementar servidores de archivos a través del protocolo FTP.

Http: es un protocolo de solicitud y respuesta a través de TCP, regularmente utilizando el puerto 80.

Iptables: Conjunto de políticas de seguridad de acceso a computadoras

Para el desarrollo de la interfaz web deberan estar instaladas las siguientes herramientas de desarrollo jdk, oc4j, jdevstudio10132.

5.4 Recurso Humano

Como equipo humano tenemos a las siguientes personas, las cuales han contribuido con profesionalismo por medio de sus conocimientos para el desarrollo de este proyecto, estas son:

Hernán Jaramillo Jiménez

Jorge Reyes Tomalá

Mónica Zúñiga Tapia

5.5 Infraestructura

La infraestructura que se utilizo para que este proyecto de desarrollara con éxito comprende lo siguiente:

- Conexión a Red
- Tarjetas de Red
- Conexión a Internet

CAPITULO 6

6 Recomendaciones y Conclusiones 85

6.1 Recomendaciones 86

6.2 Hardware 86

6.3 Software 87

6.4 Cableado 88

6.5 Puesta en Marcha 88

6.6 Conclusiones 88

6.7 Glosario de Términos 89-94

CAPITULO 6

6 Recomendaciones y Conclusiones

El capitulo trata de algunas recomendaciones para la utilización e instalación correcta del servidor de archive corporativo

6.1 Recomendaciones

El Servidor de Archivos Corporativo SACS fue creado para facilitar y dar seguridad a los datos de los usuarios dentro de un dominio. Para que pueda operar de un modo correcto y de manera óptima, deberá tener en cuenta las configuraciones de hardware, software.

Se recomienda leer detenidamente los manuales de usuario, y manual técnico del proyecto.

La implantación de SACS, facilitara el manejo de archivos compartidos y lo pueda una persona con pocos conocimientos de redes.

6.2 Hardware

Para la implementación de nuestro proyecto se utilizo un computador con las siguientes características:

Características	Servidor
Procesador	Intel Pentium IV 2.8Ghz I
Mainboard	BioStar MSI 845
Memoria RAM	512MB DDR 333Mhz
Disco Duro	80Gb

Amplitud del bus	32bits
Frecuencia del bus	533Mhz
Monitor	SVGA 15”

Las características de este computador que hizo la función de servidor debemos tomar en cuenta al momento de implantarlo en una organización que estará todo el día encendido por lo tanto se recomienda utilizar un equipo que sea propiamente un servidor ya que están fabricados para soportar trabajos exigentes y estar prendido todo el tiempo.

6.3 Software

Para implantar nuestra aplicación web en el servidor se necesita y recomienda que tenga instalado los siguientes programas.

- Sistema Operativo Linux Distribución Centos5
- instalado, configurado y funcionando samba como controlador primario.
- instalado, configurado y funcionando openldap para autenticación.
- instalado, configurado y funcionando DNS.
- instalado, configurado y funcionando smbldap-tools.
- instalado, configurado y funcionando http.
- instalado una maquina virtual java.

- instalado, configurado y funcionando las cuotas de disco.

6.4 Cableado

Se recomienda que el cableado de la red se encuentre en buen estado.

y funcionando.

6.5 Puesta en Funcionamiento

una vez pasado con éxito los puntos anteriores la puesta en funcionamiento de la aplicación web es sencilla subimos el proyecto a la carpeta o ruta creada para levantar la pagina web del proyecto.

abrimos un navegador o browser y escribimos el dominio del proyecto

ej. www.sacs.com este levantara la pagina inicial, si ocurre algún problema deberíamos ejecutar el demonio del servidor web con el que estemos trabajando sea este apache o el oc4j.

6.6 Conclusiones

La realización de este proyecto ha significado un reto importante para nosotros pero ha demostrado lo importante que es la tecnología conociendo nuevas áreas tecnológicas donde podemos encontrar un camino para desarrollarnos profesionalmente.

El desarrollo de este proyecto nos permitirá poder ayudar en una parte con las demás personas que apoyan al movimiento de software libre para que mas empresas estén seguras de poder implementar esta tecnología en sus instalaciones.

6.7 Glosario de Términos.

A

Autenticación.- Verificación de que el cliente sea quien dice ser para lograr acceso al sistema.

Administrador: Usuario con un conjunto definido de privilegios administrativos.

Aplicación: En informática, las aplicaciones son los programas con los cuales el usuario final interactúa a través de una interfaz y que realizan tareas útiles para éste.

Arquitectura: Diseño que muestra los bloques de construcción física y lógica de una aplicación distribuida (u otro sistema de software) y las relaciones que se establecen entre sí.

B

Backup.- Respaldo de información del sistema.

Base de Datos: Guarda los datos del sistema. Termino genérico que se utiliza para designar el sistema de administración de base de datos relacionales (RDBMS). Paquete de software que permite crear y manipular grandes cantidades de datos relacionados y organizados.

C

Código Fuente.- Lenguaje mediante el cual fue programado el sistema.

Conexión.- Comunicación entre varias maquinas.

Cifrado: El cifrado es el proceso de convertir el texto plano en un galimatías ilegible, denominado texto cifrado o criptograma.

Criptografía: La criptografía (del griego kryptos, "ocultar", y grafos, "escribir", literalmente "escritura oculta") es el arte o ciencia de cifrar y descifrar información utilizando técnicas matemáticas que hagan posible el intercambio de mensajes de manera que sólo puedan ser leídos por las personas a quienes van dirigidos.

E

Eth0.- Interfaz de red en Linux

Encriptar: Es convertir una cadena legible de datos en una cadena codificada de datos por medio de la clave pública , esta cadena codificada carece de significado a menos que se desencripte con la clave privada

F

Firewall.- Conjunto de políticas de seguridad de acceso a computadoras.

Freeware.- Software gratuito.

G

GPL.- Licencia pública general aplicada al software gratuito.

GUI.- Interfaces gráficas de usuario.

H

Hardware.- Parte física de una PC.

Hub.- Elemento físico de conexión en red.

I

ICMP.- Protocolo de control de mensajes de Internet

IP.- Identificador único que distingue una computadora de otra.

Interfaz.- Medio grafico de comunicación entre la PC y el usuario.

L

LAN.- Red de área local.

Logs.- Archivos temporales

M

Maquina Virtual.- Software que permite la simulación de un computador.

MYSQL.- Software para crear una base de datos.

Multiplataforma.- Se puede trabajar sobre diferentes sistemas operativos sin que afecte su desempeño.

Mensaje de Error: Mensaje que informa de un error o alguna otra situación. La finalidad de los mensajes denominados “errores de notificación” es meramente informativo.

N

O

Open Source. - Tecnología libre de distribución por la cual no se necesita licencia.

P

Passwords.- Contraseña de acceso al sistema.

Paquetes.- Segmento de datos que se transmite de un lugar a otro.

Parámetros.- Datos que se envía a una función para que cumpla con su propósito.

Periféricos.- Medios físicos del computador.

Políticas de Seguridad.- Conjunto de normas y reglas para mantener seguro un sistema.

Protocolos. – Implementación de la lógica de una capa del modelo OSI.

R

Red.- Conjunto de maquinas que se comunican entre si.

Root.- Usuario administrador

S

Servidor.- Equipo de computo, el cual esta recibiendo constante peticiones de clientes para proveerle de algún servicio.

Software.- Programas en ejecución.

Scripts. - Conjunto de líneas de códigos que permiten la ejecución de una tarea en particular.

Switch.- Dispositivo de interconexión de redes de computadoras que opera en la capa 2 del modelo OSI, este interconecta dos o más segmentos de red.

T

Telnet.- Servicio que permite abrir una interfaz de comunicación.

U

Usuario: Persona o servicio que utiliza una aplicación. Un usuario está definido de forma programática por un nombre de usuario, una contraseña y un conjunto de atributos que permiten a una aplicación reconocerlos.

V

Vulnerable.- Sensibilidad para sufrir un ataque.

.

BIBLIOGRAFIA

http://www.linuxparatodos.net

http://www.alcancelibre.com

http://www.todolinux.com

http://www.adictosaltrabajo.com

http://www.javahispano.com

ANEXO 1

Cronograma de Actividades.

ANEXO 2

Archivos de Configuraciones

2.1 DNS

2.1.1 sacs.com.zone

2.1.2 15.168.192.a

2.1.3 named.conf

2.1.4 resolv.conf

2.2 SAMBA

2.2.1 smb.conf

2.3 OPENLDAP

2.3.1 sldap.conf

2.4 SMBLDAP-TOOLS

2.4.1 /etc/smbldap-tools/smbldap_bind.conf

2.4.2 /etc/smbldap-tools/smbldap.conf

2.1 DNS

2.1.1 /var/named/chroot/var/named/sacs.com.zone

```
$TTL    86400
sacs.com.       IN SOA  server.sacs.com. root.sacs.com. (
                                        2007070601      ;serial
                                        3H              ;refresh
                                        15M             ;retry
                                        1W              ;expire
                                        1D)             ;minimun
                IN      NS      sacs
sacs            IN      A       192.168.15.254
www             IN      CNAME   sacs
```

2.1.2 /var/named/chroot/var/named/15.168.192.in-addr.arp.zone

```
$TTL 86400
@       IN SOA  server.sacs.com. root.sacs.com.    (
                                        2007070601      ;serial
                                        3H              ;refresh
                                        15M             ;retry
                                        1W              ;expire
                                        1D)             ;minium
        IN      NS              server.
254     IN      PTR             server.sacs.com.
```

2.1.3 /var/named/chroot/etc/named.conf

```
options {
        directory "/var/named";
        dump-file "/var/named/data/cache_dump.db";
      statistics-file "/var/named/data/named_stats.txt";
        forwarders {
                200.93.192.148;
                200.93.192.161;
        };
};
zone "." IN {
        type hint;
        file "named.ca";
};

zone "sacs.com" IN {
        type master;
        file "sacs.com.zone";
        allow-update {none; };
};

zone "15.168.192.in-addr.arpa" IN {
        type master;
        file "15.168.192.in-addr.arpa.zone";
        allow-update { none; };
};
```

2.1.4 resolv.conf

```
search sacs.com
nameserver 192.168.15.254
```

2.2 SAMBA

2.2.1 smb.conf

```
# This is the main Samba configuration file. You should read the
# smb.conf(5) manual page in order to understand the options listed
# here. Samba has a huge number of configurable options (perhaps too
# many!) most of which are not shown in this example
#
# For a step to step guide on installing, configuring and using samba,
# read the Samba-HOWTO-Collection. This may be obtained from:
#  http://www.samba.org/samba/docs/Samba-HOWTO-Collection.pdf
#
# Many working examples of smb.conf files can be found in the
# Samba-Guide which is generated daily and can be downloaded from:
#  http://www.samba.org/samba/docs/Samba-Guide.pdf
#
# Any line which starts with a ; (semi-colon) or a # (hash)
# is a comment and is ignored. In this example we will use a #
# for commentry and a ; for parts of the config file that you
# may wish to enable
#
# NOTE: Whenever you modify this file you should run the command
"testparm"
# to check that you have not made any basic syntactic errors.
#
#======================= Global Settings
=====================================
[global]
# workgroup = NT-Domain-Name or Workgroup-Name, eg: MIDEARTH
   workgroup = sacs

# server string is the equivalent of the NT Description field
   server string = Servidor Archivo
   netbios name = server
#############parametros LDAP###########
# Quien es el usuario administrador del dominio
  admin users = Administrador @"Domain Admins"
  passdb backend = ldapsam:ldap://192.168.15.254

#? sufijo ldap para todas las entradas siguientes
 ldap suffix = dc=sacs,dc=com

# OU de usuarios netbios
 ldap user suffix = ou=People

# OU de grupos netbios
 ldap group suffix = ou=Group
# Cuentas de maquinas netbios
 ldap machine suffix = ou=Computers

# la cuenta administrador openLDAP
 ldap admin dn = cn=Administrador,dc=sacs,dc=com

#Sincronizacion de cuentas LDAP, NT Y LM
 ldap passwd sync = yes

# agregado de cuentas maquina automaticamente
add machine script = /usr/sbin/smbldap-useradd -w %u
#### fin de parametros LDAP ##########
# Security mode. Defines in which mode Samba will  operate.
Possible
```

```
# values are share, user, server, domain and ads. Most people will want
# user level security. See the Samba-HOWTO-Collection for details.
   security = user

# This option is important for security. It allows you to restrict
# connections to machines which are on your local network. The
# following example restricts access to two C class networks and
# the "loopback" interface. For more examples of the syntax see
# the smb.conf man page
;   hosts allow = 192.168.1. 192.168.2. 127.

# If you want to automatically load your printer list rather
# than setting them up individually then you'll need this
   load printers = yes

# you may wish to override the location of the printcap file
;   printcap name = /etc/printcap

# on SystemV system setting printcap name to lpstat should allow
# you to automatically obtain a printer list from the SystemV spool
# system
;   printcap name = lpstat

# It should not be necessary to specify the print system type unless
# it is non-standard. Currently supported print systems include:
# bsd, cups, sysv, plp, lprng, aix, hpux, qnx
;   printing = cups
# This option tells cups that the data has already been rasterized
cups options = raw

# Uncomment this if you want a guest account, you must add this to
/etc/passwd
# otherwise the user "nobody" is used
;  guest account = pcguest

# this tells Samba to use a separate log file for each machine
# that connects
   log file = /var/log/samba/%m.log

# Put a capping on the size of the log files (in Kb).
   max log size = 50

# Use password server option only with security = server
# The argument list may include:
#   password server = My_PDC_Name [My_BDC_Name]
[My_Next_BDC_Name]
# or to auto-locate the domain controller/s
#   password server = *
;   password server = <NT-Server-Name>
unix password sync = Yes
passwd program = /usr/bin/passwd %u
#passwd chat = *New*UNIX*password* %n\n *ReType*new*UNIX*password*
%n\n
#*passwd:*all*authentication*tokens*updated*successfully*

# Use the realm option only with security = ads
# Specifies the Active Directory realm the host is part of
;   realm = MY_REALM

# Backend to store user information in. New installations should
# use either tdbsam or ldapsam. smbpasswd is  available for backwards
```

```
# compatibility. tdbsam requires no further configuration.
;   passdb backend = tdbsam

# Using the following line enables you to customise your configuration
# on a per machine basis. The %m gets replaced with the netbios name
# of the machine that is connecting.
# Note: Consider carefully the location in the configuration file of
       this line.  The included file is read at that point.
;   include = /usr/local/samba/lib/smb.conf.%m

# Configure Samba to use multiple interfaces
# If you have multiple network interfaces then you must list them
# here. See the man page for details.
;   interfaces = 192.168.12.2/24 192.168.13.2/24
interfaces = 192.168.15.254/24
# Browser Control Options:
# set local master to no if you don't want Samba to become a master
# browser on your network. Otherwise the normal election rules apply
   local master = yes

# OS Level determines the precedence of this server in master browser
# elections. The default value should be reasonable
   os level = 65

# Domain Master specifies Samba to be the Domain Master Browser. This
# allows Samba to collate browse lists between subnets. Don't use this
# if you already have a Windows NT domain controller doing this job
   domain master = yes

# Preferred Master causes Samba to force a local browser election on startup
# and gives it a slightly higher chance of winning the election
   preferred master = yes

# Enable this if you want Samba to be a domain logon server for
# Windows95 workstations.
   domain logons = yes

# if you enable domain logons then you may want a per-machine or
# per user logon script
# run a specific logon batch file per workstation (machine)
;   logon script = %m.bat
# run a specific logon batch file per username
;   logon script = %U.bat
    logon script = logon.cmd
# Where to store roving profiles (only for Win95 and WinNT)
#       %L substitutes for this servers netbios name, %U is username
#       You must uncomment the [Profiles] share below
   logon path = \\%L\Profiles\%U
   logon home = \\%L\%U\.profile
   logon drive = H:

# Windows Internet Name Serving Support Section:
# WINS Support - Tells the NMBD component of Samba to enable it's WINS
Server
;   wins support = yes

# WINS Server - Tells the NMBD components of Samba to be a WINS Client
#           Note: Samba can be either a WINS Server, or a WINS Client, but NOT
both
;   wins server = w.x.y.z
```

```
# WINS Proxy - Tells Samba to answer name resolution queries on
# behalf of a non WINS capable client, for this to work there must be
# at least one       WINS Server on the network. The default is NO.
;   wins proxy = yes

# DNS Proxy - tells Samba whether or not to try to resolve NetBIOS names
# via DNS nslookups. The default is NO.
   dns proxy = no

# These scripts are used on a domain controller or stand-alone
# machine to add or delete corresponding unix accounts
;  add user script = /usr/sbin/useradd %u
;  add group script = /usr/sbin/groupadd %g
;  add machine script = /usr/sbin/adduser -n -g machines -c Machine -d /dev/null
-s /bin/false %u
;  delete user script = /usr/sbin/userdel %u
;  delete user from group script = /usr/sbin/deluser %u %g
;  delete group script = /usr/sbin/groupdel %g


#================Share Definitions ==============================
[homes]
   comment = Home Directories
   browseable = no
   writable = yes

# Un-comment the following and create the netlogon directory for Domain
Logons

 [netlogon]
       comment = Network Logon Service
   path = /var/lib/samba/netlogon
   guest ok = yes
   write list = @administradores, @admins_dominio
   browseable = yes

# Un-comment the following to provide a specific roving profile share
# the default is to use the user's home directory

[Profiles]
    path = /var/lib/samba/profiles
    read only = no
    guest ok = yes
    create mask = 0600
    directory mask = 0700
# NOTE: If you have a BSD-style print system there is no need to
# specifically define each individual printer
[printers]
   comment = All Printers
   path = /usr/spool/samba

browseable = no
# Set public = yes to allow user 'guest account' to print
   guest ok = no
   writable = no
   printable = yes
# This one is useful for people to share files
;[tmp]
;   comment = Temporary file space
;   path = /tmp
```

```
;   read only = no
;   public = yes

[tmp]
   comment = sambacompart
   path = /tmp
# A publicly accessible directory, but read only, except for people in
# the "staff" group
;[public]
;   comment = Public Stuff
;   path = /home/samba
;   public = yes
;   writable = yes
;   printable = no
;   write list = @staff

# Other examples.
#
# A private printer, usable only by fred. Spool data will be placed in fred's
# home directory. Note that fred must have write access to the spool directory,
# wherever it is.
;[fredsprn]
;   comment = Fred's Printer
;   valid users = fred
;   path = /homes/fred
;   printer = freds_printer
;   public = no
;   writable = no
;   printable = yes

# A private directory, usable only by fred. Note that fred requires write
# access to the directory.
;[fredsdir]
;   comment = Fred's Service
;   path = /usr/somewhere/private
;   valid users = fred
;   public = no
;   writable = yes
;   printable = no

# a service which has a different directory for each machine that connects
# this allows you to tailor configurations to incoming machines. You could
# also use the %U option to tailor it by user name.
# The %m gets replaced with the machine name that is connecting.
;[pchome]
;  comment = PC Directories
;  path = /usr/pc/%m
;  public = no
;  writable = yes

# A publicly accessible directory, read/write to all users. Note that all files
# created in the directory by users will be owned by the default user, so
# any user with access can delete any other user's files. Obviously this
# directory must be writable by the default user. Another user could of course
# be specified, in which case all files would be owned by that user instead.

;[public]
;   path = /usr/somewhere/else/public
;   public = yes
;   only guest = yes
;   writable = yes
;   printable = no
```

```
# The following two entries demonstrate how to share a directory so that two
# users can place files there that will be owned by the specific users. In this
# setup, the directory should be writable by both users and should have the
# sticky bit set on it to prevent abuse. Obviously this could be extended to
# as many users as required.
;[myshare]
;   comment = Mary's and Fred's stuff
;   path = /usr/somewhere/shared
;   valid users = mary fred
;   public = no
;   writable = yes
;   printable = no
;   create mask = 0765

[contabilidad2]
comment = Archivos comunes de contabilidad2
path = /home/samba/contabilidad2
valid users = @contabilidad2
write list = @contabilidad2
force group = contabilidad2
create mode = 770
directory mode = 770
[produccion]
comment = Archivos comunes de produccion
path = /home/samba/produccion
valid users = @produccion
write list = @produccion
force group = produccion
create mode = 770
directory mode = 770
[Recaudacion]
comment = Archivos comunes de Recaudacion
path = /home/samba/Recaudacion
valid users = @Recaudacion
write list = @Recaudacion
force group = Recaudacion
create mode = 770
directory mode = 770
```

2.3 openLdap

2.3.1 sladp.conf

```
#
# See slapd.conf(5) for details on configuration options.
# This file should NOT be world readable.
#
include         /etc/openldap/schema/core.schema
include         /etc/openldap/schema/cosine.schema
include         /etc/openldap/schema/inetorgperson.schema
include         /etc/openldap/schema/nis.schema
include         /etc/openldap/schema/samba.schema
# Allow LDAPv2 client connections.  This is NOT the default.
allow bind_v2

# Do not enable referrals until AFTER you have a working directory
# service AND an understanding of referrals.
#referral ldap://root.openldap.org

pidfile         /var/run/openldap/slapd.pid
argsfile   /var/run/openldap/slapd.args

# Load dynamic backend modules:
# modulepath    /usr/lib/openldap
# moduleload    back_bdb.la
# moduleload    back_ldap.la
# moduleload    back_ldbm.la
# moduleload    back_passwd.la
# moduleload    back_shell.la

# The next three lines allow use of TLS for encrypting connections using a
# dummy test certificate which you can generate by changing to
# /etc/pki/tls/certs, running "make slapd.pem", and fixing permissions on
# slapd.pem so that the ldap user or group can read it.  Your client software
# may balk at self-signed certificates, however.
# TLSCACertificateFile /etc/pki/tls/certs/ca-bundle.crt
# TLSCertificateFile /etc/pki/tls/certs/slapd.pem
# TLSCertificateKeyFile /etc/pki/tls/certs/slapd.pem

# Sample security restrictions
#       Require integrity protection (prevent hijacking)
#       Require 112-bit (3DES or better) encryption for updates
#       Require 63-bit encryption for simple bind
# security ssf=1 update_ssf=112 simple_bind=64

# Sample access control policy:
#       Root DSE: allow anyone to read it
#       Subschema (sub)entry DSE: allow anyone to read it
#Other DSEs:
#               Allow self write access
#               Allow authenticated users read access
#               Allow anonymous users to authenticate
#       Directives needed to implement policy:
# access to dn.base="" by * read
# access to dn.base="cn=Subschema" by * read
# access to *
#       by self write
#       by users read
#       by anonymous auth
```

```
#
# if no access controls are present, the default policy
# allows anyone and everyone to read anything but restricts
# updates to rootdn.  (e.g., "access to * by * read")
#
# rootdn can always read and write EVERYTHING!

#######################################################################
#
# ldbm and/or bdb database definitions
#######################################################################
#

database        bdb
suffix          "dc=sacs,dc=com"
rootdn          "cn=Administrador,dc=sacs,dc=com"
# Cleartext passwords, especially for the rootdn, should
# be avoided.  See slappasswd(8) and slapd.conf(5) for details.
# Use of strong authentication encouraged.
# rootpw        secret
rootpw      {SSHA}b8JF2tZ+ST+w5tWM+s42MafnJaUMaXQN
# The database directory MUST exist prior to running slapd AND
# should only be accessible by the slapd and slap tools.
# Mode 700 recommended.
directory/var/lib/ldap/autenticar

# Indices to maintain for this database
#index objectClass                      eq,pres
#index ou,cn,mail,surname,givenname     eq,pres,sub
#index uidNumber,gidNumber,loginShell    eq,pres
#index uid,memberUid                    eq,pres,sub
#index nisMapName,nisMapEntry           eq,pres,sub
index objectClass                       eq,pres
index cn                                eq,pres,sub
index sn                                eq,pres,sub
index uid                               eq,pres,sub
index displayName                       eq,pres,sub
index uidNumber                         eq
index gidNumber                         eq
index memberUID                         eq
index sambaSID                          eq
index sambaPrimaryGroupSID              eq
index sambaDomainName                   eq
index default                           sub
# Replicas of this database
#replogfile /var/lib/ldap/openldap-master-replog
#replica host=ldap-1.example.com:389 starttls=critical
#     bindmethod=sasl saslmech=GSSAPI
#     authcId=host/ldap-master.example.com@EXAMPLE.COM
```

2.4 smbldap-tools

2.4.1 /etc/smbldap-tools/smbldap_bind.conf

```
##############################
# Credential Configuration #
##############################
# Notes: you can specify two differents configuration if you use a
# master ldap for writing access and a slave ldap server for reading access
# By default, we will use the same DN (so it will work for standard Samba
# release)
slaveDN="cn=Administrador,dc=sacs,dc=com"
slavePw="emelec"
masterDN="cn=Administrador,dc=sacs,dc=com"
masterPw="emelec"
```

2.4.2 /etc/smbldap-tools/smbldap.conf

```
# $Source: $
# $Id: smbldap.conf,v 1.18 2005/05/27 14:28:47 jtournier Exp $
#
# smbldap-tools.conf : Q & D configuration file for smbldap-tools

#  This code was developped by IDEALX (http://IDEALX.org/) and
#  contributors (their names can be found in the CONTRIBUTORS file).
#
#               Copyright (C) 2001-2002 IDEALX
#
#  This program is free software; you can redistribute it and/or
#  modify it under the terms of the GNU General Public License
#  as published by the Free Software Foundation; either version 2
#  of the License, or (at your option) any later version.
#
#  This program is distributed in the hope that it will be useful,
#  but WITHOUT ANY WARRANTY; without even the implied warranty of
#  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
#  GNU General Public License for more details.
#
#  You should have received a copy of the GNU General Public License
#  along with this program; if not, write to the Free Software
#  Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307,
#  USA.

#  Purpose :
#       . be the configuration file for all smbldap-tools scripts

##############################################################################
#
# General Configuration
#
##############################################################################

# Put your own SID. To obtain this number do: "net getlocalsid".
```

```
# If not defined, parameter is taking from "net getlocalsid" return
SID="S-1-5-21-4233777812-4109637176-3988859424"

# Domain name the Samba server is in charged.
# If not defined, parameter is taking from smb.conf configuration file
# Ex: sambaDomain="sacs"
sambaDomain="sacs"

##############################################################################
#
# LDAP Configuration
#
##############################################################################

# Notes: to use to dual ldap servers backend for Samba, you must patch
# Samba with the dual-head patch from IDEALX. If not using this patch
# just use the same server for slaveLDAP and masterLDAP.
# Those two servers declarations can also be used when you have
# . one master LDAP server where all writing operations must be done
# . one slave LDAP server where all reading operations must be done
#   (typically a replication directory)

# Slave LDAP server
# Ex: slaveLDAP=127.0.0.1
# If not defined, parameter is set to "127.0.0.1"
slaveLDAP="192.168.15.254"

# Slave LDAP port
# If not defined, parameter is set to "389"
slavePort="389"

# Master LDAP server: needed for write operations
# Ex: masterLDAP=127.0.0.1
# If not defined, parameter is set to "127.0.0.1"
masterLDAP="192.168.15.254"

# Master LDAP port
# If not defined, parameter is set to "389"
masterPort="389"
# Use TLS for LDAP
# If set to 1, this option will use start_tls for connection
# (you should also used the port 389)
# If not defined, parameter is set to "1"
##ldapTLS="1"
# How to verify the server's certificate (none, optional or require)
# see "man Net::LDAP" in start_tls section for more details
#verify="optional"

# CA certificate
# see "man Net::LDAP" in start_tls section for more details
##cafile="/etc/smbldap-tools/ca.pem"

# certificate to use to connect to the ldap server
# see "man Net::LDAP" in start_tls section for more details
##clientcert="/etc/smbldap-tools/smbldap-tools.pem"

# key certificate to use to connect to the ldap server
# see "man Net::LDAP" in start_tls section for more details
##clientkey="/etc/smbldap-tools/smbldap-tools.key"
# LDAP Suffix
# Ex: suffix=dc=IDEALX,dc=ORG
suffix="dc=sacs,dc=com"

# Where are stored Users
# Ex: usersdn="ou=Users,dc=IDEALX,dc=ORG"
```

```
# Warning: if 'suffix' is not set here, you must set the full dn for usersdn
usersdn="ou=People,${suffix}"

# Where are stored Computers
# Ex: computersdn="ou=Computers,dc=IDEALX,dc=ORG"
# Warning: if 'suffix' is not set here, you must set the full dn for computersdn
computersdn="ou=Computers,${suffix}"

# Where are stored Groups
# Ex: groupsdn="ou=Groups,dc=IDEALX,dc=ORG"
# Warning: if 'suffix' is not set here, you must set the full dn for groupsdn
groupsdn="ou=Group,${suffix}"

# Where are stored Idmap entries (used if samba is a domain member server)
# Ex: groupsdn="ou=Idmap,dc=IDEALX,dc=ORG"
# Warning: if 'suffix' is not set here, you must set the full dn for idmapdn
idmapdn="ou=Idmap,${suffix}"

# Where to store next uidNumber and gidNumber available for new users and groups
# If not defined, entries are stored in sambaDomainName object.
# Ex: sambaUnixIdPooldn="sambaDomainName=${sambaDomain},${suffix}"
# Ex: sambaUnixIdPooldn="cn=NextFreeUnixId,${suffix}"
sambaUnixIdPooldn="sambaDomainName=${sacs},${suffix}"

# Default scope Used
scope="sub"

# Unix password encryption (CRYPT, MD5, SMD5, SSHA, SHA, CLEARTEXT)
hash_encrypt="SSHA"

# if hash_encrypt is set to CRYPT, you may set a salt format.
# default is "%s", but many systems will generate MD5 hashed
# passwords if you use "$1$%.8s". This parameter is optional!
crypt_salt_format="%s"

##############################################################################
#
# Unix Accounts Configuration
#
##############################################################################

# Login defs
# Default Login Shell
# Ex: userLoginShell="/bin/bash"
userLoginShell="/bin/bash"

# Home directory
# Ex: userHome="/home/%U"
userHome="/home/%U"

# Default mode used for user homeDirectory
userHomeDirectoryMode="700"

# Gecos
userGecos="System User"

# Default User (POSIX and Samba) GID
defaultUserGid="513"

# Default Computer (Samba) GID
defaultComputerGid="515"

# Skel dir
skeletonDir="/etc/skel"
```

```
# Default password validation time (time in days) Comment the next line if
# you don't want password to be enable for defaultMaxPasswordAge days (be
# careful to the sambaPwdMustChange attribute's value)
defaultMaxPasswordAge="9999"

################################################################################
#
# SAMBA Configuration
#
################################################################################

# The UNC path to home drives location (%U username substitution)
# Just set it to a null string if you want to use the smb.conf 'logon home'
# directive and/or disable roaming profiles
# Ex: userSmbHome="\\PDC-SMB3\%U"
userSmbHome="\\server\%U"

# The UNC path to profiles locations (%U username substitution)
# Just set it to a null string if you want to use the smb.conf 'logon path'
# directive and/or disable roaming profiles
# Ex: userProfile="\\server\profiles\%U"
userProfile="\\server\profiles\%U"

# The default Home Drive Letter mapping
# (will be automatically mapped at logon time if home directory exist)
# Ex: userHomeDrive="H:"
userHomeDrive="H:"

# The default user netlogon script name (%U username substitution)
# if not used, will be automatically username.cmd
# make sure script file is edited under dos
# Ex: userScript="startup.cmd" # make sure script file is edited under dos
userScript="logon.cmd"

# Domain appended to the users "mail"-attribute
# when smbldap-useradd -M is used
# Ex: mailDomain="idealx.com"
mailDomain="sacs.com"

################################################################################
#
# SMBLDAP-TOOLS Configuration (default are ok for a RedHat)
#
################################################################################

# Allows not to use smbpasswd (if with_smbpasswd == 0 in smbldap_conf.pm) but
# prefer Crypt::SmbHash library
with_smbpasswd="0"
smbpasswd="/usr/bin/smbpasswd"

# Allows not to use slappasswd (if with_slappasswd == 0 in smbldap_conf.pm)
# but prefer Crypt:: libraries
with_slappasswd="0"
slappasswd="/usr/sbin/slappasswd"

# comment out the following line to get rid of the default banner
# no_banner="1"
```

UNIVERSIDAD DE GUAYAQUIL

Facultad de Ciencias Matemáticas y Física

Carrera de Ingeniería en Sistemas Computacionales

“Servidor de Archivos Corporativo”

TOMO II

PROYECTO DE GRADO

Previo a la Obtención del Título de:

INGENIERO EN SISTEMAS COMPUTACIONALES

Autores:

Hernán Jaramillo Jiménez

Jorge Reyes Tomalá

Mónica Azucena Zúñiga Tapia

GUAYAQUIL – ECUADOR

Año: 2008

AGRADECIMIENTO

Agradezco de todo corazón a Jehová, por su guía y bendición y por haber permitido que llegue a la culminación de mis estudios, también a mi madre por su apoyo constante y a mis compañeros Hernán y Jorge por el compañerismo y el esfuerzo que entregaron para la realización de este proyecto.

Mónica

Agradezco a Dios por darme fuerza, protección y poder entregarle mi logro en vida a mis padres ya que gracias a su enseñanza oculta demostrando la fuerza que tienen para salir delante de las adversidades de la vida

Hernán

Agradecimiento enorme a mi Dios Todopoderoso, a mi Familia, en especial a mi madre que siempre estuvo ahí junto a mi en todo momento, a mis amigos por el apoyo constante que me han brindado para cumplir este objetivo.

Jorge

DEDICATORIA

Dedico este proyecto que representa una larga trayectoria de mi vida llena de esfuerzo, dedicación y estudios a mi madre que es la persona que ha estado siempre a mi lado con la ayuda de Jehová apoyándome día a día y este es el resultado que siempre soñé con entregarle como premio a su amor y dedicación desinteresada.

Mónica

Dedico este proyecto a mis dos hermosas gardenias Sasha y Damaris que son la inspiración diaria para salir adelante profesionalmente. Dedico el proyecto también a mi querida esposa por el gran apoyo que me ha brindado.

Hernán

Dedico este proyecto, al amor eterno de mi vida Mercedes Tomalá, mi querida madre, pilar fundamental de mi existencia, que aunque no la pueda tener presente físicamente porque Dios así lo quiso, se que esta conmigo en cada instante, en mi mente y corazón, para ti Mamá, te amo y te extraño mucho, que Dios te tenga en su gloria.

Jorge

TRIBUNAL DE GRADUACIÓN

Presidente del Tribunal	Primer Vocal
Segundo Vocal	Secretario

ÍNDICE GENERAL

AGRADECIMIENTO I

DEDICATORIA II

TRIBUNAL DE GRADUACIÓN IV

ÍNDICE GENERAL V

ÍNDICE GRÁFICO IX

Capitulo 1
MANUAL TECNICO.

1.1 Antes de Empezar 2

1.2 Instalación centOS 2 - 19

1.3 Configuración del Servidor 20

1.3.1 DNS 20

1.3.1.1 Archivo de configuración named.conf 20

1.3.1.2 Archivos de zona de resolución inversa 15.168.192.in-addr.arpa.zone 21

1.3.1.3 Zona de reenvió del dominio sacs.com.zone 21

1.3.1.4 Archivo resolv.conf 22

1.3.2 Configuración de Samba 22

1.3.2.1 Paquetes necesarios para la configuración de Samba 23

1.3.2.2 Fichero lmhosts 24

1.3.2.3 smb.conf 24

1.3.2.4 Prueba del Dominio 27

1.3.3 Configuración OpenLDAP 35
1.3.4 Integración Samba y OpenLDAP 38

1.3.5 Instalación y configuración de smbldap-tools 42

1.3.6 Configuración de la base de datos 49

1.3.7 Configuración del Virtual Hosts 49

1.3.8 Configuración de VSFTP 50

1.4 Codificación
59

1.4.1 Clases Principales 59

1.4.1.1 Grupo.java 59

1.4.1.2 Usuario.java 62

1.4.1.3 Perfil.java 64

1.4.1.4 UsuarioPerfil.java 64

1.4.1.5 Jftp.java 66

1.4.1.6 Conexión.java 68

1.4.2 JSP Principales 69

1.4.2.1 jabrirArchivo.jsp 69

1.4.2.2 jaccionAbrirArchivo.jsp 73

1.4.2.3 jaccionAsignarCuotaG.jsp 75

1.4.2.4 jaccionCambiarClaveU.jsp 76

1.5 Scripts 78

1.5.1 SCrearUsuario 78

1.5.2 SCrearGrupo 79

1.5.3 ScrearPermiso 79

1.5.4 SCrearGrpSamba 80

1.5.5 SAsignarGrupo 80

1.5.6 SClave 81

1.5.7 SDesabilitarUsuario 81

1.5.8 SDesabilitarGrupo 81

1.5.9 SAsignarCuotaUsuario 82

1.5.10 SAsignarCuotaGrupo 82

Capitulo 2
MANUAL DE USUARIO

2.1 Pantalla Principal 83

2.2 Inicio de Sesión 88

2.3 Menú Administrador 89

2.3.1 Administración de Usuarios. 89

2.3.1.1Nuevo Usuario 90

2.3.1.2 Cambiar clave 91

2.3.1 3 Activar Usuario. 92

2.3.1.4 Inactivar Usuario 93

2.3.1.5 Asignar Grupo 94

2.3.1.6 Cuota 95

2.3.1 7 Consultar 96

2.3.2 Administración Grupos de Usuario 97

2.3.2.1 Nuevo 98

2.3.2.2 Activar Grupo 99

2.3.2.3 Inactivar Grupo 100

2.3.2.4 cuotas 101

2.3.2.5 Consultar 102

2.3.3 Administración Archivos 103

2.3.3.1 Abrir Directorio 104

2.3.3.2 Abrir Archivo 105

2.3.3.3 Liberar 106

2.3.3.4 Historial 107

2.3.3.5 Respaldar 108

2.4 Menú Usuario 109

2.4 .1 Menú Archivos 110

2.4.1.1 Historial 111

2.4.1.2 Antivirus 112

ÍNDICE GRÁFICO

Figura 1.1 Modo Instalacion centOS 2
Figura 1.2 Verificacion de CD's de Instalacion 3
Figura 1.3 Inicio de Instalacion CentOS 3
Figura 1.4 Selección de Idioma de Instalacion 4
Figura 1.5 Idioma del Teclado 4
Figura 1.6 Detecion de Instalaciones Anteriores de CentOS 5
Figura 1.7 Seleccionar Particion Personalizada 5
Figura 1.8 Espacio libre del Disco 6
Figura 1.9 Asignacion de espacio memoria swap 7
Figura 1.10 Resumen Particiones Creadas 7
Figura 1.11 Gestor de Arranque 8
Figura 1.12 Configuracion Parametros de Red 9
Figura 1.13 Modificacion Parametros de Red 9
Figura 1.14 Configuracion Horario 10
Figura 1.15 Asignacion Clave deRoot 11
Figura 1.16 Instalacion de Paquetes 12
Figura 1.17 Inico de progreso de Instalacion centOS 12
Figura 1.18 Progreso de Instalacion centOS 13
Figura 1.19 Finalizacion de Instalacion centOS 13
Figura 1.20 Inicio del Sistema Operativo 14
Figura 1.21 Pantalla de Bienvenido de CentOS 14
Figura 1.22 Activacion/Desactivacion de Cortafuegos 15
Figura 1.23 Creacion de Usuario 16
Figura 1.24 Autenticacion de Usuario 16
Figura 1.25 Configuracion Tarjeta de Sonido 17
Figura 1.26 Ingreso de Usuario 17
Figura 1.27 Ingreso de Contraseña 18
Figura 1.28 Escritorio de centOS 18
Figura 1.29 Pantalla de Grub 19
Figura 1.30 Arbol LDAP 35
Figura 1.31 Configuración de la Autenticación. 38

CAPITULO 1

MANUAL TECNICO

Este capitulo esta dedicado para personal que desee implementar el servidor de archivos corporativos. Aquí detallaremos los archivos de configuración los scripts, la codificación. Además se detalla paso a paso los procesos realizados para implementar esta aplicación y configurar el servidor.

1.1 Antes de Empezar

Se recomienda leer primero este apartado del manual, con respecto a la instalación y configuración del sistema para su correcto uso y funcionamiento.

1.2 INSTALACION DE CENTOS

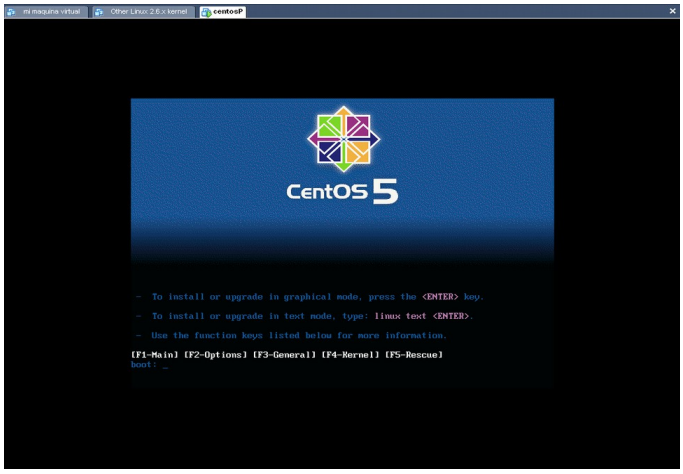

mi maquina virtual
Other Linux 2.6.x kernel
centosP
CentOS 5
- To install or upgrade in graphical mode, press the <ENTER> key.
- To install or upgrade in text mode, type: linux text <ENTER>.
- Use the function keys listed below for more information.
[F1-Main] [F2-Options] [F3-General] [F4-Kernel] [F5-Rescue]
boot: _

Figura 1.1 Modo instalación centOS

Para realizar la instalación de CentOS, insertamos el cd de instalación cuando aparezca el cuadro de dialogo de inicio boot pulse la tecla ENTER para iniciar la instalación en modo grafico.

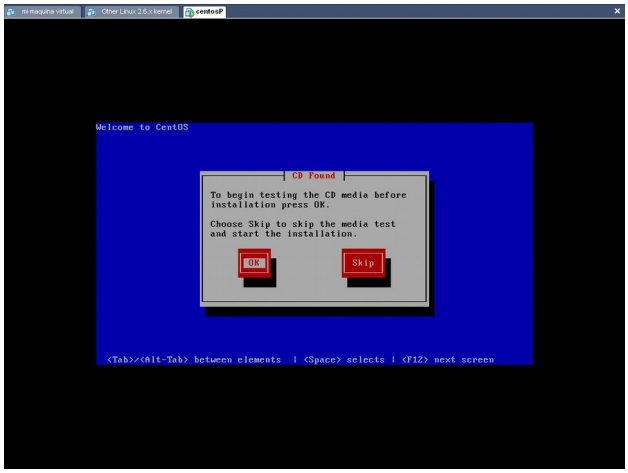


Figura 1.2 Verificación de CD's de Instalación

Si desea verificar la integridad del cd pulse la tecla OK y pulse la tecla ENTER y espere varios minutos, si esta seguro de que el disco o los discos se encuentran en buen estado seleccione SKIP y pulse ENTER.

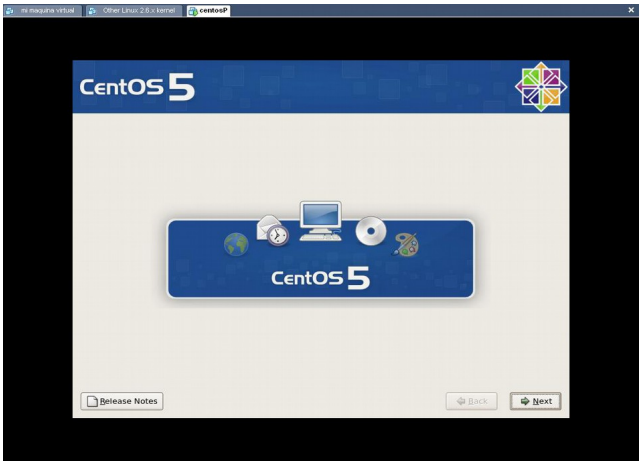


Figura 1.3 Inicio de Instalación CentOS

Pulse la tecla ENTER en la pantalla de bienvenida al programa de instalación de CentOS

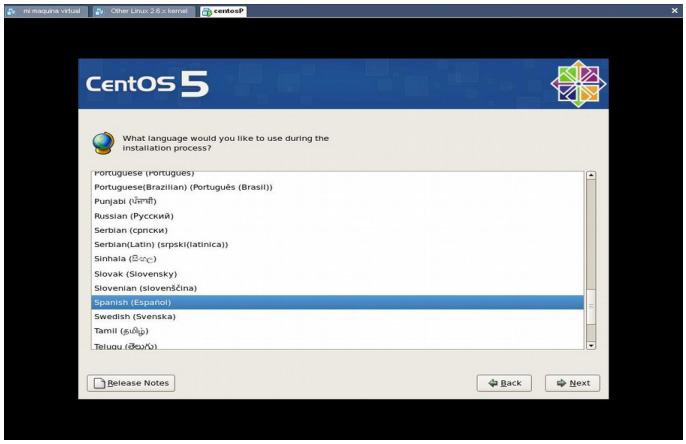


Figura 1.4 Selección de Idioma De Instalación.

Seleccione Spanish como idioma para ser utilizado en la instalación

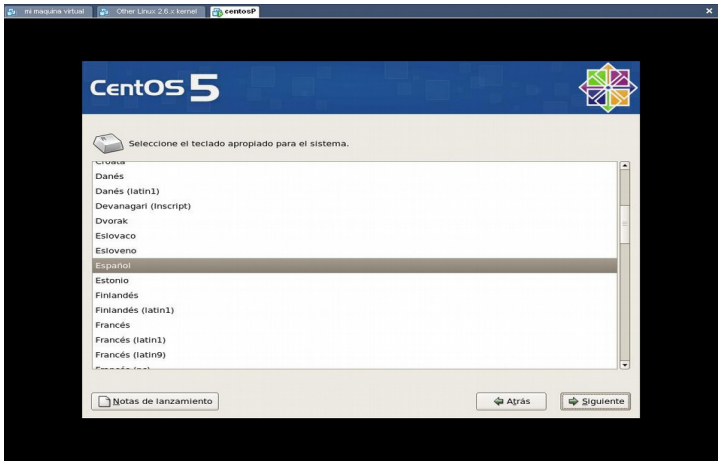


Figura 1.5 Idioma del Teclado

Seleccione el mapa de teclado que corresponda al dispositivo utilizado. El mapa **«Spanish»** corresponde a la disposición del teclado Español España. Al terminar, haga clic sobre el botón **«Siguiente»**.

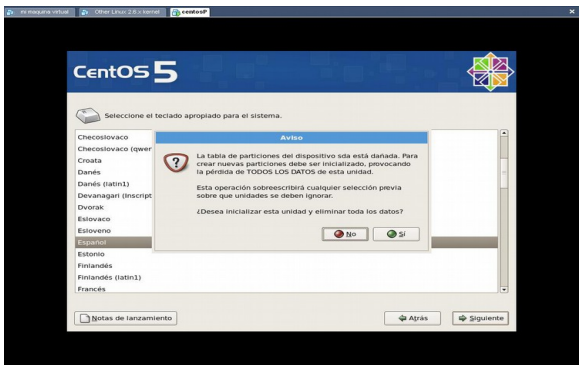


Figura 1.6 Detección de Instalaciones Anteriores de CentOS

Haga clic sobre el botón **«Siguiente»** y espere a que el sistema intente detectar instalaciones previas de CentOS. Al aviso que se muestra en la pantalla debemos dar click en el botón si.

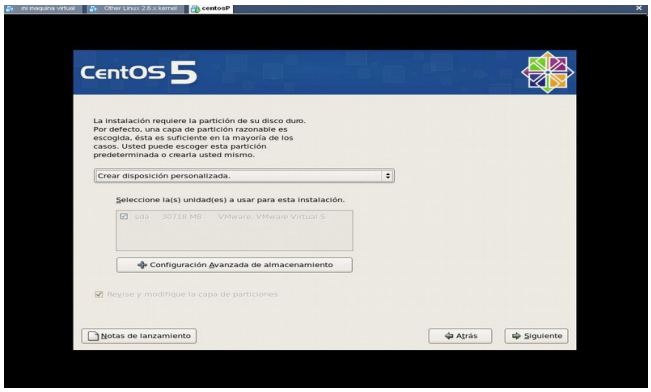

CentOS 5
La instalación requiere la partición de su disco duro. Por defecto, una capa de partición razonable es escogida, ésta es suficiente en la mayoría de los casos. Usted puede escoger esta partición predeterminada o crearla usted mismo.
Crear disposición personalizada.
Seleccione la(s) unidad(es) a usar para esta instalación.
sda 30718 MB VMware, VMware Virtual S
Configuración avanzada de almacenamiento
Revise y modifique la capa de particiones
Notas de lanzamiento
Atrás
Siguiente

Figura 1.7 Seleccionar Partición Personalizada

Dando clic en la flecha del combo con la leyenda remover particiones botón configuración avanzada de almacenamiento para crear las particiones buscamos la opción crear disposiciones avanzadas.

Damos click en **<<Siguiente>>**

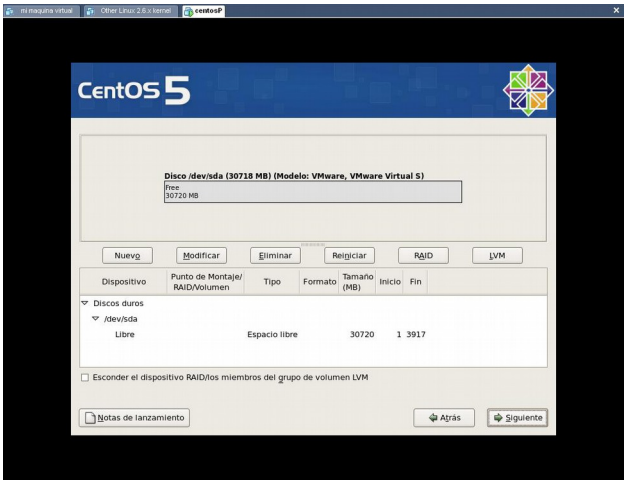


Figura 1.8 Espacio libre del disco.

LA pantalla nos muestra el espacio libre del disco duro.

Haga clic en el botón **«Nuevo»**. Para comenzar a crear las particiones.

Cree las particiones y asígnele los siguientes valores:

- 100 MB a la partición /boot.
- 15000MB a la partición del /home.
- 3000 a la partición /tmp.
- 3000 al directorio raiz /.
- 100 al /Boot
- 3000 a la partición /var

- 3000 al /swap.

Luego damos valores a la memoria SWAP como lo muestra la siguiente pantalla.

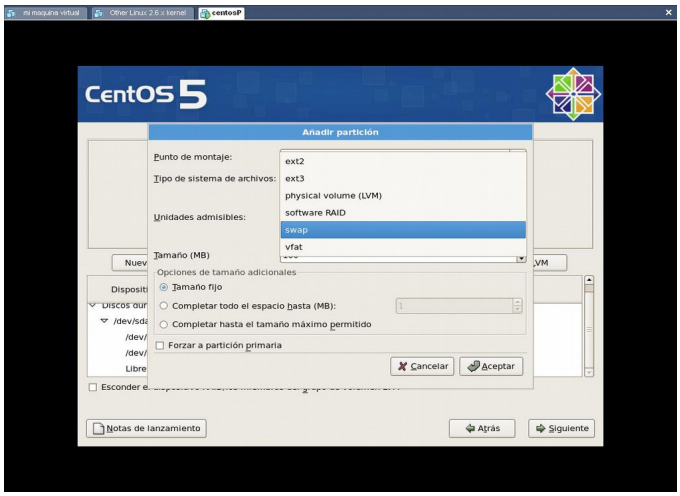


Figura 1.9 Asignación de espacio memoria swap

La siguiente pantalla muestra un resumen de las particiones creadas.

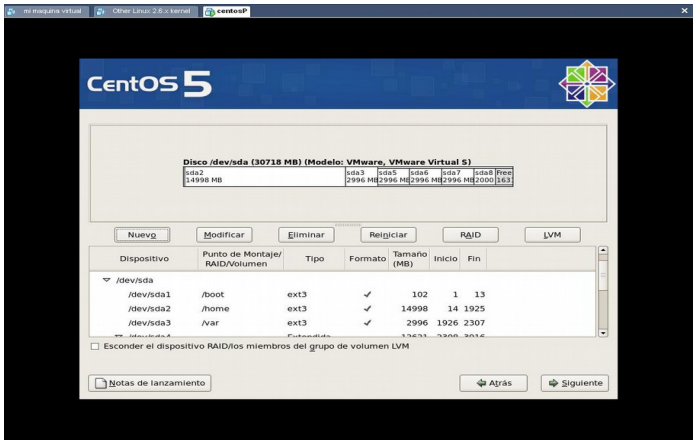


Figura 1.10 Resumen Particiones Creadas.

Si está conforme con la tabla de particiones creada haga clic sobre el botón **«siguiente»** para pasar a la siguiente pantalla.

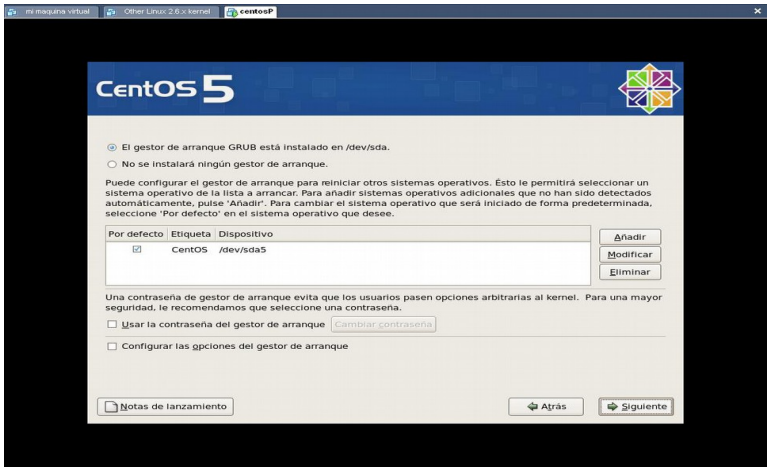

mi maquina virtual
Other Linux 2.6.x kernel
centosP
CentOS 5
El gestor de arranque GRUB está instalado en /dev/sda.
No se instalará ningún gestor de arranque.
Puede configurar el gestor de arranque para reiniciar otros sistemas operativos. Esto le permitirá seleccionar un sistema operativo de la lista a arrancar. Para añadir sistemas operativos adicionales que no han sido detectados automáticamente, pulse 'Añadir'. Para cambiar el sistema operativo que será iniciado de forma predeterminada, seleccione 'Por defecto' en el sistema operativo que desee.
Por defecto
Etiqueta
Dispositivo
CentOS
/dev/sda5
Añadir
Modificar
Eliminar
Una contraseña de gestor de arranque evita que los usuarios pasen opciones arbitrarias al kernel. Para una mayor seguridad, le recomendamos que seleccione una contraseña.
Usar la contraseña del gestor de arranque
Cambiar contraseña
Configurar las opciones del gestor de arranque
Notas de lanzamiento
Atrás
Siguiente

Figura 1.11 Gestor de Arranque.

Esta pantalla nos indica donde esta instalada el gestor de arranque.

También podemos dar click en la casilla **«Usar la contraseña del gestor de arranque».** Con la finalidad de impedir que alguien sin autorización y con acceso físico al sistema pueda iniciar el sistema en nivel de corrida 1, o cualquiera otro, haga clic.

Se abrirá una ventana emergente donde deberá ingresar, con confirmación, la clave de acceso exclusiva para el gestor de arranque. Al terminar, haga clic sobre el botón **«Aceptar»**

Al terminar, haga clic sobre el botón **«Siguiente»**.

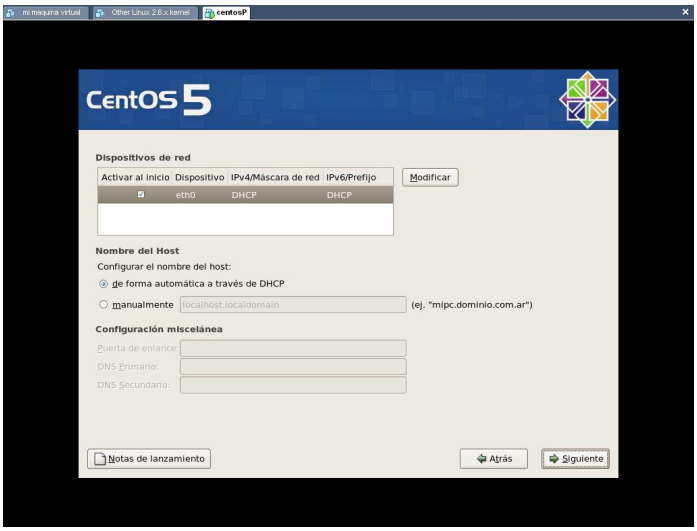


Figura 1.12 Configuración Parámetros de Red.

Para configurar los parámetros de red del sistema, haga clic sobre el botón **«Modificar»** para la interfaz eth0.

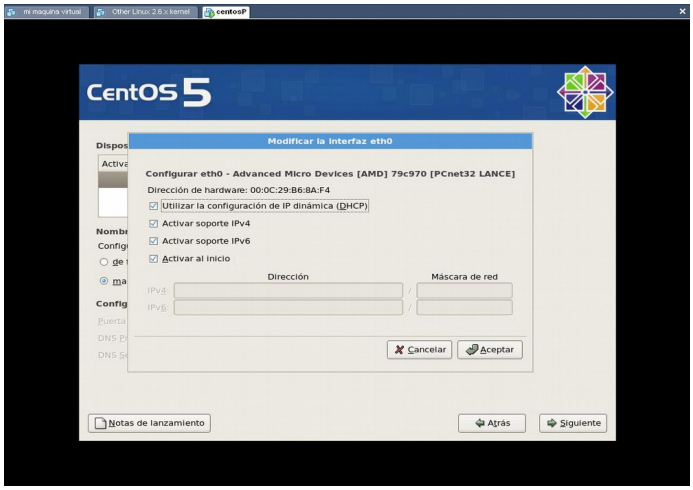


Figura 1.13 Modificación parámetros de Red

En la ventana emergente para modificar la interfaz eth0, desactive la casilla **«Configurar usando DHCP»** y especifique la dirección IP y máscara de subred que utilizará en adelante el sistema. Al terminar haga clic en Aceptar.

Asigne un nombre de anfitrión (HOSTNAME) para el sistema. Se recomienda que sea un **FQDN** resuelto al menos en un DNS local. Defina la dirección IP de la puerta de enlace y las direcciones IP de los servidores DNS. Al terminar, haga clic sobre el botón **«Siguiente»**.

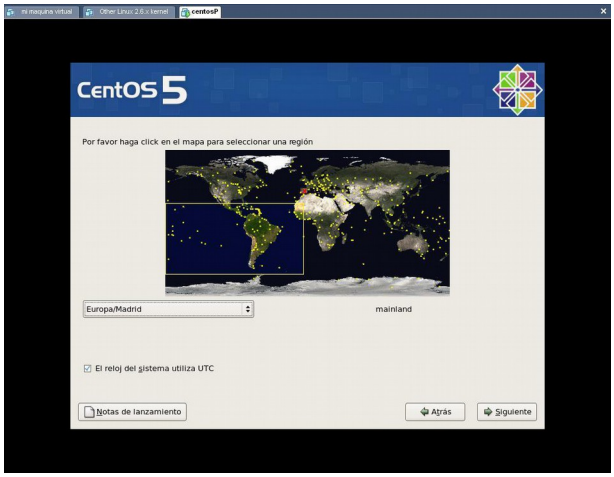


Figura 1.14 Configuración Horario

Seleccione la casilla **«El sistema horario usará UTC»**, que significa que el reloj del sistema utilizará **UTC** (**T**iempo **U**niversal **C**oordinado), Haga clic con el ratón sobre la región que corresponda en el mapa mundial

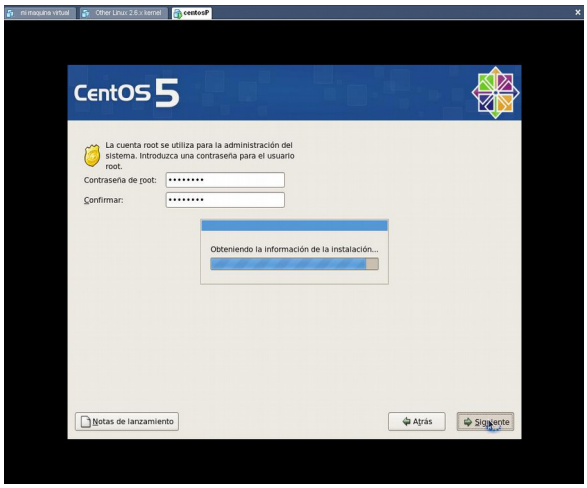

CentOS 5
La cuenta root se utiliza para la administración del sistema. Introduzca una contraseña para el usuario root.
Contraseña de root:
Confirmar:
Obteniendo la información de la instalación...
Notas de lanzamiento
Atrás
Siguiente

Figura 1.15 Asignación clave de Root

Asigne una clave de acceso al usuario **root**. Debe escribirla dos veces a fin de verificar que está coincide con lo que realmente se espera. Por razones de seguridad, se recomienda asignar una clave de acceso que evite utilizar palabras provenientes de cualquier diccionario, en cualquier idioma, así como cualquier combinación que tenga relación con datos personales. De clic en **<<Siguiente>>**

En la siguiente pantalla podrá seleccionar los grupos de paquetes que quiera instalar en el sistema.

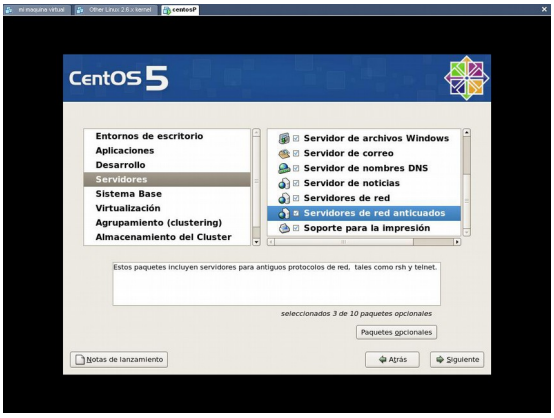


Figura 1.16 Instalación de Paquetes

La ventana muestra del lado derecho un menú y al dar clic sobre este muestra del lado izquierdo todos los paquetes disponibles para instalar seleccionamos dando un clic en la casilla del paquete que queremos instalar y damos clic en **«Siguiente»**.

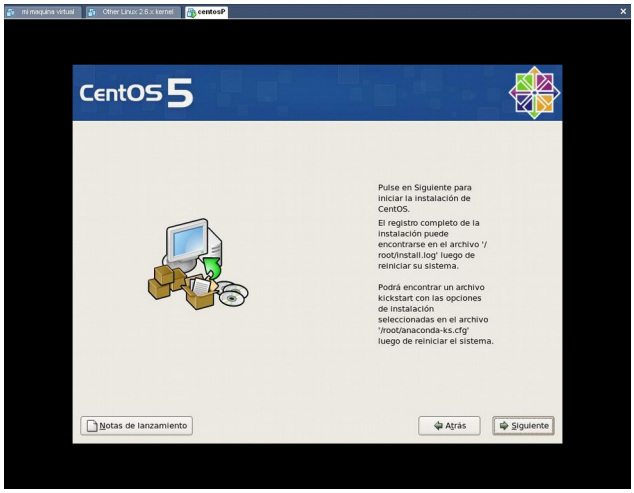

mi maquina virtual
Other Linux 2.6.x kernel
centosP
CentOS 5
Pulse en Siguiente para iniciar la instalación de CentOS.
El registro completo de la instalación puede encontrarse en el archivo '/root/install.log' luego de reiniciar su sistema.
Podrá encontrar un archivo kickstart con las opciones de instalación seleccionadas en el archivo '/root/anaconda-ks.cfg' luego de reiniciar el sistema.
Notas de lanzamiento
Atrás
Siguiente

Figura 1.17 Inicio de Progreso de Instalación centOS.

Damos clic en siguiente para empezar la instalación de CentOS

Figura 1.18 Progreso de Instalacion centOS

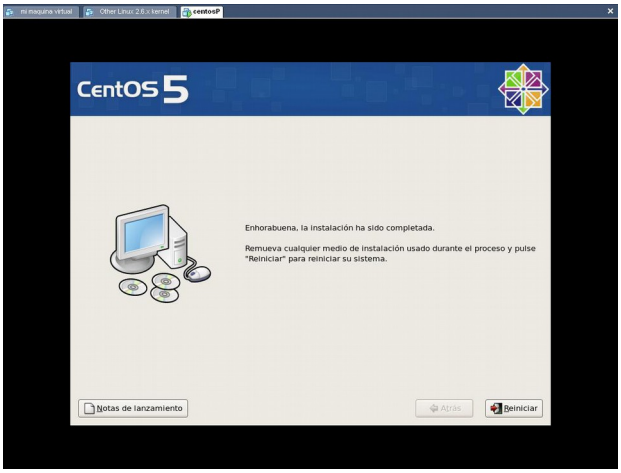

mi maquina virtual
Other Linux 2.6.x kernel
centosP
CentOS 5
Enhorabuena, la instalación ha sido completada.
Remueva cualquier medio de instalación usado durante el proceso y pulse "Reiniciar" para reiniciar su sistema.
Notas de lanzamiento
Atrás
Reiniciar

Figura 1.19 Finalización de Instalación centOS

Este mensaje sale cuando ya termina la instalación de CentOS y damos clic en “reiniciar” para reiniciar el sistema.

Luego aparece la siguiente pantalla en donde esta iniciando el Sistema Operativo CentOS.

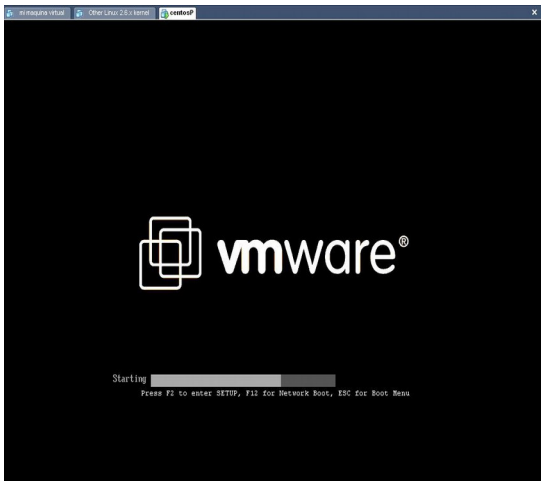


Figura 1.20 Inicio del Sistema Operativo

Aparecerá la pantalla de bienvenido al sistema CentOS

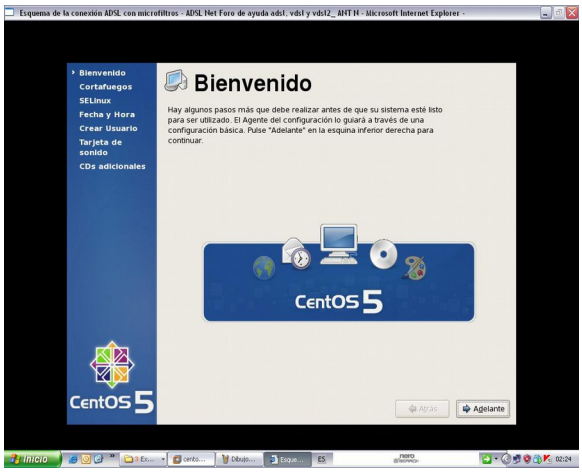

Bienvenido
Cortafuegos
SELinux
Fecha y Hora
Crear Usuario
Tarjeta de sonido
CDs adicionales
Bienvenido
Hay algunos pasos más que debe realizar antes de que su sistema esté listo para ser utilizado. El Agente del configuración lo guiará a través de una configuración básica. Pulse "Adelante" en la esquina inferior derecha para continuar.
CentOS 5
Atrás
Adelante
CentOS 5

Figura 1.21 Pantalla de Bienvenido de centOS

Deshabilitamos el cortafuego y clic en siguiente, luego nos aparecerá una ventana informando que se sobrescribirán el archivo de firewall, aceptamos.

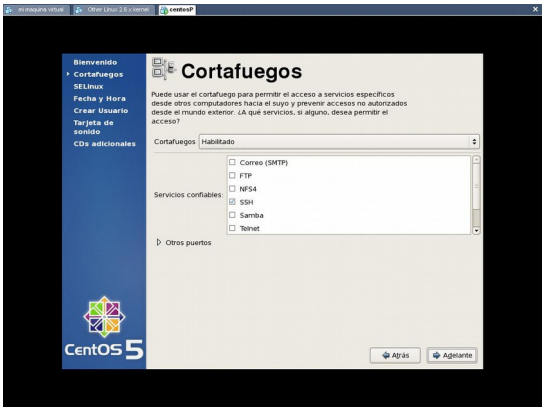


Figura 1.22 Activación/desactivación de Cortafuego

Escogemos la opción obediente

Creamos un usuario llamado server con su respectiva contraseña

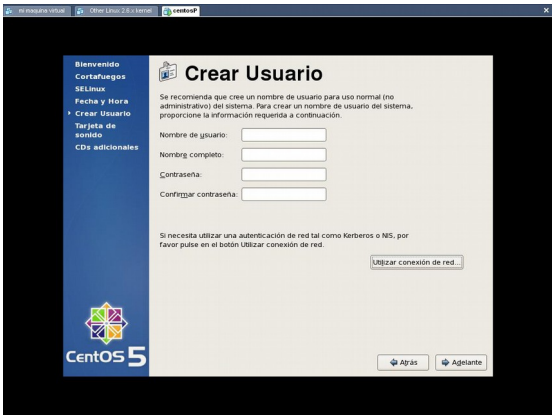

Bienvenido
Cortafuegos
SELinux
Fecha y Hora
Crear Usuario
Tarjeta de sonido
CDs adicionales
Crear Usuario
Se recomienda que cree un nombre de usuario para uso normal (no administrativo) del sistema. Para crear un nombre de usuario del sistema, proporcione la información requerida a continuación.
Nombre de usuario:
Nombre completo:
Contraseña:
Confirmar contraseña:
Si necesita utilizar una autenticación de red tal como Kerberos o NIS, por favor pulse en el botón Utilizar conexión de red.
Utilizar conexión de red...
CentOS 5
Atrás
Adelante

Figura 1.23 Creación de Usuario

En esta pantalla se realiza la configuración de la autenticación de usuario.

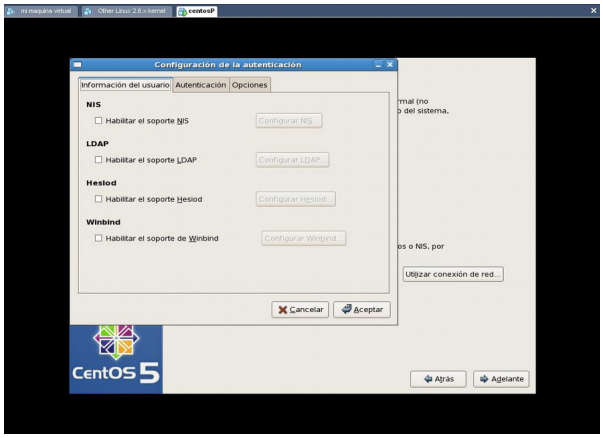


Figura 1.24 Autenticación de Usuario

En esta pantalla se configura el volumen del dispositivo de audio.

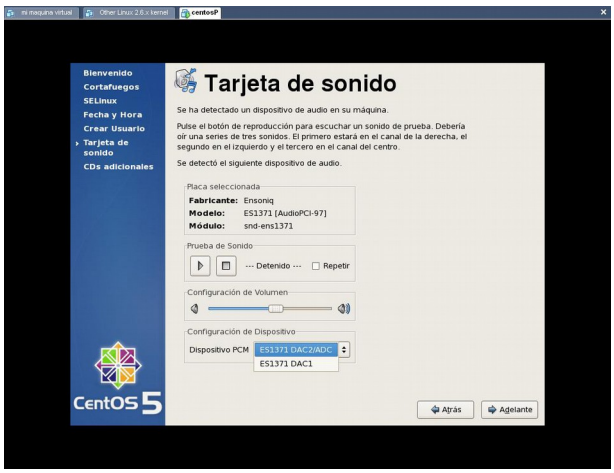

Bienvenido
Cortafuegos
SELinux
Fecha y Hora
Crear Usuario
Tarjeta de sonido
CDs adicionales
Tarjeta de sonido
Se ha detectado un dispositivo de audio en su máquina.
Pulse el botón de reproducción para escuchar un sonido de prueba. Debería oír una series de tres sonidos. El primero estará en el canal de la derecha, el segundo en el izquierdo y el tercero en el canal del centro.
Se detectó el siguiente dispositivo de audio.
Placa seleccionada
Fabricante: Ensoniq
Modelo: ES1371 [AudioPCI-97]
Módulo: snd-ens1371
Prueba de Sonido
Detenido
Repetir
Configuración de Volumen
Configuración de Dispositivo
Dispositivo PCM
ES1371 DAC2/ADC
ES1371 DAC1
CentOS 5
Atrás
Adelante

Figura 1.25 Configuración de Tarjeta de Sonido

Esta es la pantalla de ingreso al sistema operativo, para ingresar se debe digitar su nombre de usuario.

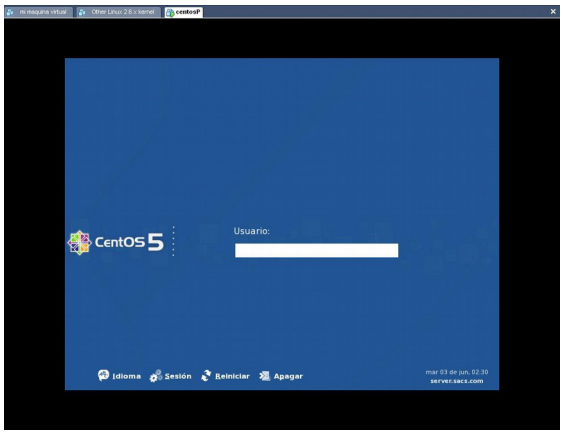


Figura 1.26 Ingreso de Usuario

Ingresamos la clave del usuario antes ingresado

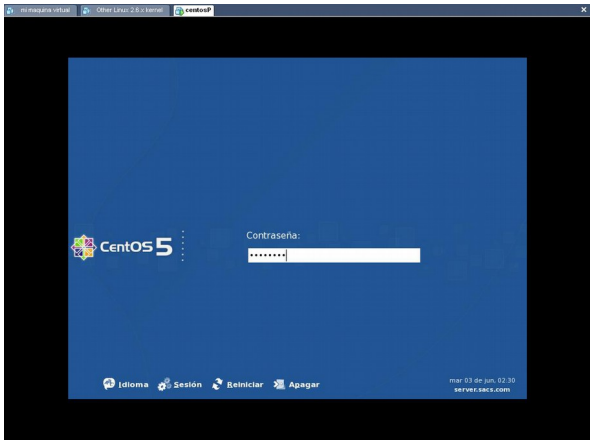


Figura 1.27 Ingreso de Contraseña

En esta pantalla podemos visualizar el escritorio del Sistema Operativo CentOS

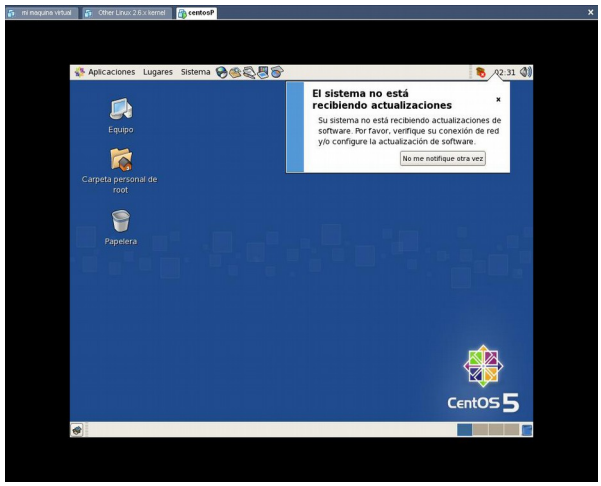

Aplicaciones
Lugares
Sistema
El sistema no está recibiendo actualizaciones
Su sistema no está recibiendo actualizaciones de software. Por favor, verifique su conexión de red y/o configure la actualización de software.
No me notifique otra vez
Equipo
Carpeta personal de root
Papelera
CentOS 5

Figura 1.28 Escritorio de centOS

Existen situaciones en las cuales se puede requerir el inicio del sistema en nivel de corrida 1 o nivel mono-usuario a fin de realizar tareas de mantenimiento o, en su defecto, reparaciones.

Al iniciar el sistema, éste lo hará presentando la pantalla del gestor de arranque conocido como Grub presentando una pantalla similar a la siguiente:

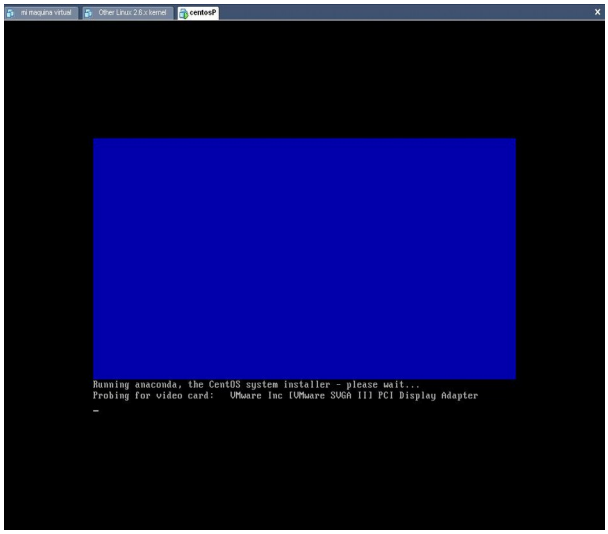

mi maquina virtual
Other Linux 2.6.x kernel
centosP
Running anaconda, the CentOS system installer - please wait...
Probing for video card: VMware Inc [VMware SVGA II] PCI Display Adapter

Figura 1.29 Pantalla de Grub

1.3 Configuración del Servidor

1.3.1 DNS

El Domain Name System (DNS) o sistema de nombres de dominios, es una base de datos que almacena información asociada a nombres de dominio.

1.3.1.1 Archivo de Configuración named.conf

Este archivo permite establecer el nombre del dominio, definir las zonas de reenvió y la zona inversa para la resolución de nombres de nuestro dominio.

```
options {

        directory "/var/named";
        dump-file "/var/named/data/cache_dump.db";
    statistics-file "/var/named/data/named_stats.txt";
        forwarders {
                200.93.192.148;
                200.93.192.161;
        };
};
zone "." IN {
        type hint;
        file "named.ca";
};
zone "15.168.192.in-addr.arpa" IN {
        type master;
        file "15.168.192.in-addr.arpa.zone";
        allow-update { none; };
};
zone "sacs.com" IN {
        type master;
        file "sacs.com.zone";
        allow-update {none; };
};
```

1.3.1.2 Archivos de zona de resolución inversa 15.168.192.in-addr.arpa.zone

Este archivo permite configurar la IP de nuestro servidor y poder así realizar una búsqueda usando la dirección IP para que este a su vez me resuelva y me de el nombre del dominio.

var/named/chroot/var/named/15.168.192.in-addr.arpa.zone

```
$TTL 86400
@       IN SOA  server.sacs.com. root.sacs.com.    (
                                        2007070601      ;serial
                                        3H              ;refresh
                                        15M             ;retry
                                        1W              ;expire
                                        1D)             ;minium
@       IN      NS              server.
254     IN      PTR             server.sacs.com.
```

1.3.1.3 Zona de reenvió del dominio sacs.com.zone

Este archivo es el más importante ya que define el archivo de configuración name.ca el cual indica como contactar a otros dominios, esta en la ruta:

/var/named/chroot/var/named/sacs.com.zone

```
$TTL    86400
@       IN SOA  server.sacs.com. root.sacs.com. (
                                        2007070601      ;serial
                                        3H              ;refresh
                                        15M             ;retry
                                        1W              ;expire
                                        1D)             ;minimun
@               IN      NS      server.
@               IN      MX      10      mail
server          IN      A       192.168.15.254
mail            IN      A       192.168.15.254
ftp             IN      A       192.168.15.254
www             IN      CNAME server
```

1.3.1.4 Archivo resolv.conf

Este archivo es utilizado por el DNS para pode buscar el servidor ya sea por el nombre o la dirección IP que resolverá los nombres de la red local se encuentra en la ruta:

/etc/resolv.conf

search sacs.com

nameserver 192.168.15.254

1.3.2 Configuración de Samba

SAMBA es un conjunto de programas, bajo la Licencia Publica General GNU, y que se implementan en sistemas basados sobre UNIX® el protocolo SMB. Sirve como reemplazo total para Windows NT, NFS o servidores Netware.

1.3.2.1 Paquetes necesarios para la configuración de samba:

• samba: Servidor SMB.

• samba-client: Diversos clientes para el protoclo SMB.

• samba-common: Ficheros necesarios para cliente y servidor.

Instalamos los paquetes necesarios con el comando

rpm -q samba.rpm.

Debemos crear un usuario el mismo que tendrá que estar creado tanto en Windows como en Linux con los siguientes comandos:

useradd -s /sbin/nologin **hernan**

smbpasswd -a **hernan**

useradd -s /bin/bash **hernan** //crea usuario windows

passwd **hernan** //password para usuario windows

smbpasswd -a **hernan** // añado usuario a samba.

1.3.2.2 El fichero lmhosts :

En este fichero debemos ingresar las direcciones IP con sus respectivos nombres para poder resolver por el netbios.

127.0.0.1 localhost

192.168.15.10 estacion.sacs.com

1.3.2.3 smb.conf

smb.conf es el archivo principal de configuración para samba podemos modificar este archivo de configuración con cualquier editor de texto.

Tenemos que establecer un grupo de trabajo, editando el valor del parámetro workgroup.

workgroup **= sacs** //Nombre del Dominio.

netbios name **= server** // Nombre de la Maquina servidor

server string **= servidor de archivo //**

Hosts allow Permite establecer la lista de control de acceso que definirá que maquinas o redes podrán acceder hacia nuestro servidor.

Interfaces Permite establecer que interfaces de red del sistema se escucharan peticiones.

interface: **192.168.15.0/24**

domain master = Yes //

domain logons = Yes //accesos por dominios actuando como PDC

logon path = \\%L\Profiles\%U.

logon home = \\%L\%U\.profile

logon drive = H: // establece la letra de unidad en un cliente NT

logon script = logon.cmd //se ejecutara en el cliente una vez que el usuario se haya conectado al dominio

El fichero /var/lib/samba/netlogon/logon.cmd deberá contener algo como lo siguiente:

REM windows client logon script

REM

net time \\mi-servidor /SET /YES

net use H: \\mi-servidor\homes /PERSISTENT:NO

Definimos los guiones a ejecutar para distintas tareas como alta de máquinas, usuarios y grupos así como la baja de éstos.

add user script = /usr/sbin/useradd %u

add machine script = /usr/sbin/useradd -d /dev/null -g 100 -s /bin/false -c

"Cuenta de máquina" -M %u

delete user script = **/usr/sbin/userdel %u**

delete group script = **/usr/sbin/groupdel %g**

add user to group script = **/usr/bin/gpasswd -a %u %g**

set primary group script = **/usr/sbin/usermod -g %g %u**

Definimos los recursos donde residirá netlogon y también dónde se almacenarán los perfiles de los usuarios:

[netlogon]

comment = Network Logon Service

path = /var/lib/samba/netlogon

write list = @administradores, @admins_dominio

guest ok = Yes

browseable = Yes

[Profiles]

path = /var/lib/samba/profiles

read only = No

guest ok = Yes

create mask = 0600

directory mask = 0700

Creamos las carpetas netlogon y profiles en la siguiente ruta:

Damos permiso a las carpetas.

mkdir -p -m 1777 /var/lib/samba/profiles

mkdir -p /var/lib/samba/netlogon

chgrp users /var/lib/samba/profiles

Para iniciar Samba por primera vez, utilice:

#service smb start

Si va a reiniciar el servicio, realice lo siguiente:

#service smb restart

Para que Samba inicie automáticamente cada vez que inicie el servidor sólo ejecute el siguiente mandato:

#chkconfig smb on

1.3.2.5 Prueba del Dominio

Debemos primero tener conectado en red nuestro cliente Windows.

Debemos tener presente que para cada versión del sistema operativo de Windows existen diferentes formas de conectar a un dominio.

En nuestro proyecto utilizamos un cliente con sistema operativo Windows XP : debemos realizar los siguientes pasos :

- Clic derecho en el icono de «Mi PC».
- Seleccionar «Propiedades».
- Haga clic en la pestaña de «Identificación de red» o «Nombre del sistema».
- Clic en el botón de «Propiedades».
- Clic en el botón «Miembro de dominio».
- Ingrese el nombre del dominio y el nombre de la máquina y haga clic en el botón de «Aceptar».
- Aparecerá un diálogo que preguntará por una cuenta y clave de acceso con privilegios de administración en el servidor.

  Especifique el usuario root y la clave de acceso que asignó a la cuenta de root con el mandato smbpasswd (**NO LA CLAVE DE ACCESO DE ROOT EN EL SISTEMA**).
- Espere algunos segundos.
- Deberá mostrarse un mensaje emergente de confirmación que

dice «Bienvenido a MI-DOMINIO».

• Reinicie el sistema.

• Acceda con cualquier usuario que haya sido dado de alta en el servidor Samba y que además cuente con una clave de acceso asignada con smbpasswd

Archivo /etc/samba/smb.conf

```
# This is the main Samba configuration file. You should read the
# smb.conf(5) manual page in order to understand the options listed
# here. Samba has a huge number of configurable options (perhaps too
# many!) most of which are not shown in this example
#
# For a step to step guide on installing, configuring and using samba,
# read the Samba-HOWTO-Collection. This may be obtained from:
#  http://www.samba.org/samba/docs/Samba-HOWTO-Collection.pdf
#
# Many working examples of smb.conf files can be found in the
# Samba-Guide which is generated daily and can be downloaded from:
#  http://www.samba.org/samba/docs/Samba-Guide.pdf
#
# Any line which starts with a ; (semi-colon) or a # (hash)
# is a comment and is ignored. In this example we will use a #
# for commentry and a ; for parts of the config file that you
# may wish to enable
#
# NOTE: Whenever you modify this file you should run the command "testparm"
# to check that you have not made any basic syntactic errors.
#
#======================= Global Settings =====================================
[global]

# workgroup = NT-Domain-Name or Workgroup-Name, eg: MIDEARTH
   workgroup = sacs

# server string is the equivalent of the NT Description field
   server string = Servidor Archivo
   netbios name = server
##############parametros LDAP############
# Quien es el usuario administrador del dominio
  admin users = Administrador @"Domain Admins"
  passdb backend = ldapsam:ldap://192.168.15.254

#? sufijo ldap para todas las entradas siguientes
 ldap suffix = dc=sacs,dc=com

# OU de usuarios netbios
 ldap user suffix = ou=People

# OU de grupos netbios
 ldap group suffix = ou=Group

# Cuentas de maquinas netbios
 ldap machine suffix = ou=Computers

# la cuenta administrador openLDAP
 ldap admin dn = cn=Administrador,dc=sacs,dc=com

#Sincronizacion de cuentas LDAP, NT Y LM
 ldap passwd sync = yes

# agregado de cuentas maquina automaticamente
add machine script = /usr/sbin/smbldap-useradd -w %u
#### fin de parametros LDAP ##########
# Security mode. Defines in which mode Samba will operate. Possible
# values are share, user, server, domain and ads. Most people will want
# user level security. See the Samba-HOWTO-Collection for details.
   security = user

# This option is important for security. It allows you to restrict
# connections to machines which are on your local network. The
# following example restricts access to two C class networks and
# the "loopback" interface. For more examples of the syntax see
# the smb.conf man page
```

```
;   hosts allow = 192.168.1. 192.168.2. 127.

# If you want to automatically load your printer list rather
# than setting them up individually then you'll need this
   load printers = yes

# you may wish to override the location of the printcap file
;   printcap name = /etc/printcap

# on SystemV system setting printcap name to lpstat should allow
# you to automatically obtain a printer list from the SystemV spool
# system
;   printcap name = lpstat

# It should not be necessary to specify the print system type unless
# it is non-standard. Currently supported print systems include:
# bsd, cups, sysv, plp, lprng, aix, hpux, qnx
;   printing = cups

# This option tells cups that the data has already been rasterized
cups options = raw

# Uncomment this if you want a guest account, you must add this to /etc/passwd
# otherwise the user "nobody" is used
;  guest account = pcguest

# this tells Samba to use a separate log file for each machine
# that connects
   log file = /var/log/samba/%m.log

# Put a capping on the size of the log files (in Kb).
   max log size = 50

# Use password server option only with security = server
# The argument list may include:
#   password server = My_PDC_Name [My_BDC_Name] [My_Next_BDC_Name]
# or to auto-locate the domain controller/s
#   password server = *
;   password server = <NT-Server-Name>
unix password sync = Yes
passwd program = /usr/bin/passwd %u
#passwd chat = *New*UNIX*password* %n\n *ReType*new*UNIX*password* %n\n
#*passwd:*all*authentication*tokens*updated*successfully*

# Use the realm option only with security = ads
# Specifies the Active Directory realm the host is part of
;   realm = MY_REALM

# Backend to store user information in. New installations should
# use either tdbsam or ldapsam. smbpasswd is available for backwards
# compatibility. tdbsam requires no further configuration.
;   passdb backend = tdbsam

# Using the following line enables you to customise your configuration
# on a per machine basis. The %m gets replaced with the netbios name
# of the machine that is connecting.
# Note: Consider carefully the location in the configuration file of
       this line.  The included file is read at that point.
;   include = /usr/local/samba/lib/smb.conf.%m

# Configure Samba to use multiple interfaces
# If you have multiple network interfaces then you must list them
# here. See the man page for details.
;   interfaces = 192.168.12.2/24 192.168.13.2/24
interfaces = 192.168.15.254/24
# Browser Control Options:
# set local master to no if you don't want Samba to become a master
```

```
# browser on your network. Otherwise the normal election rules apply
   local master = yes

# OS Level determines the precedence of this server in master browser
# elections. The default value should be reasonable
   os level = 65

# Domain Master specifies Samba to be the Domain Master Browser. This
# allows Samba to collate browse lists between subnets. Don't use this
# if you already have a Windows NT domain controller doing this job
   domain master = yes

# Preferred Master causes Samba to force a local browser election on startup
# and gives it a slightly higher chance of winning the election
   preferred master = yes

# Enable this if you want Samba to be a domain logon server for
# Windows95 workstations.
   domain logons = yes

# if you enable domain logons then you may want a per-machine or
# per user logon script
# run a specific logon batch file per workstation (machine)
;   logon script = %m.bat
# run a specific logon batch file per username
;   logon script = %U.bat
    logon script = logon.cmd
# Where to store roving profiles (only for Win95 and WinNT)
#        %L substitutes for this servers netbios name, %U is username
#        You must uncomment the [Profiles] share below
   logon path = \\%L\Profiles\%U
   logon home = \\%L\%U\.profile
   logon drive = H:

# Windows Internet Name Serving Support Section:
# WINS Support - Tells the NMBD component of Samba to enable it's WINS Server
;   wins support = yes

# WINS Server - Tells the NMBD components of Samba to be a WINS Client
#       Note: Samba can be either a WINS Server, or a WINS Client, but NOT both
;   wins server = w.x.y.z

# WINS Proxy - Tells Samba to answer name resolution queries on
# behalf of a non WINS capable client, for this to work there must be
# at least one WINS Server on the network. The default is NO.
;   wins proxy = yes

# DNS Proxy - tells Samba whether or not to try to resolve NetBIOS names
# via DNS nslookups. The default is NO.
   dns proxy = no
# These scripts are used on a domain controller or stand-alone
# machine to add or delete corresponding unix accounts
;  add user script = /usr/sbin/useradd %u
;  add group script = /usr/sbin/groupadd %g
;  add machine script = /usr/sbin/adduser -n -g machines -c Machine -d /dev/null
-s /bin/false %u
;  delete user script = /usr/sbin/userdel %u
;  delete user from group script = /usr/sbin/deluser %u %g
;  delete group script = /usr/sbin/groupdel %g

#============================ Share Definitions ==============================
[homes]
   comment = Home Directories
   browseable = no
   writable = yes
```

```
# Un-comment the following and create the netlogon directory for Domain Logons
; [netlogon]
;   comment = Network Logon Service
;   path = /usr/local/samba/lib/netlogon
;   guest ok = yes
;   writable = no
;   share modes = no

 [netlogon]
   comment = Network Logon Service
   path = /var/lib/samba/netlogon
   guest ok = yes
   write list = @administradores, @admins_dominio
   browseable = yes

# Un-comment the following to provide a specific roving profile share
# the default is to use the user's home directory
;[Profiles]
;    path = /usr/local/samba/profiles
;    browseable = no
;    guest ok = yes

[Profiles]
    path = /var/lib/samba/profiles
    read only = no
    guest ok = yes
    create mask = 0600
    directory mask = 0700
# NOTE: If you have a BSD-style print system there is no need to
# specifically define each individual printer
[printers]
   comment = All Printers
   path = /usr/spool/samba
   browseable = no
# Set public = yes to allow user 'guest account' to print
   guest ok = no
   writable = no
   printable = yes

# This one is useful for people to share files
;[tmp]
;   comment = Temporary file space
;   path = /tmp
;   read only = no
;   public = yes

[tmp]
   comment = sambacompart
   path = /tmp
# A publicly accessible directory, but read only, except for people in
# the "staff" group
;[public]
;   comment = Public Stuff
;   path = /home/samba
;   public = yes
;   writable = yes
;   printable = no
;   write list = @staff
# Other examples.
#
# A private printer, usable only by fred. Spool data will be placed in fred's
# home directory. Note that fred must have write access to the spool directory,
# wherever it is.
;[fredsprn]
;   comment = Fred's Printer
;   valid users = fred
```

```
;   path = /homes/fred
;   printer = freds_printer
;   public = no
;   writable = no
;   printable = yes

# A private directory, usable only by fred. Note that fred requires write
# access to the directory.
;[fredsdir]
;   comment = Fred's Service
;   path = /usr/somewhere/private
;   valid users = fred
;   public = no
;   writable = yes
;   printable = no

# a service which has a different directory for each machine that connects
# this allows you to tailor configurations to incoming machines. You could
# also use the %U option to tailor it by user name.
# The %m gets replaced with the machine name that is connecting.
;[pchome]
;  comment = PC Directories
;  path = /usr/pc/%m
;  public = no
;  writable = yes

# A publicly accessible directory, read/write to all users. Note that all files
# created in the directory by users will be owned by the default user, so
# any user with access can delete any other user's files. Obviously this
# directory must be writable by the default user. Another user could of course
# be specified, in which case all files would be owned by that user instead.
;[public]
;   path = /usr/somewhere/else/public
;   public = yes
;   only guest = yes
;   writable = yes
;   printable = no

# The following two entries demonstrate how to share a directory so that two
# users can place files there that will be owned by the specific users. In this
# setup, the directory should be writable by both users and should have the
# sticky bit set on it to prevent abuse. Obviously this could be extended to
# as many users as required.
;[myshare]
;   comment = Mary's and Fred's stuff
;   path = /usr/somewhere/shared
;   valid users = mary fred
;   public = no
;   writable = yes
;   printable = no
;   create mask = 0765
[Administrador]
comment = Archivos comunes de Administrador
path = /home/samba/Administrador
valid users = @Administrador
write list = @Administrador
force group = Administrador
create mode = 770
directory mode = 770
[operadores]
comment = Archivos comunes de operadores
path = /home/samba/operadores
valid users = @operadores
write list = @operadores
force group = operadores
create mode = 770
directory mode = 770
```

1.3.3 Configuración OpenLDAP

El servidor openLdap nos permitirá crear contenedores de información para tenerlas almacenadas y poderlas administrar de una manera mas eficiente y segura.

Lo primero es definir la estructura o árbol directorio Ldap. Comenzamos con las bases para nuestro árbol:

dc=sacs, dc=com

El “dc” representa componente de dominio esto lleva un orden en nuestro proyecto la dirección web termina en .com por eso la base de nuestro árbol es dc=com seguido del nombre del dominio dc=sasc.

Luego formamos las unidades organizacionales “ou” en el proyecto manejamos 3 unidades organizacionales ou=Users, ou=Groups, ou=Computers nuestro árbol Ldap quedaria:

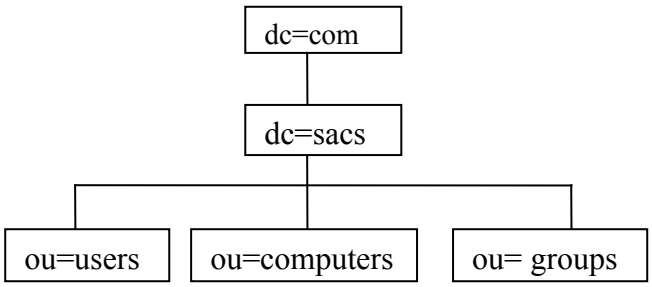


Figura 1.30 Arbol Ldap

Para implementar un servidor openldap debemos instalar los siguientes paquetes:

openldap-2.3.27-5.i386.rpm

openldap-clients-2.3.27-5.i386.rpm

openldap-servers-2.3.27-5.i386.rpm

authconfig-5.3.12-2.el5.i386.rpm

authconfig-gtk-5.3.12-2.el5.i386.rpm

utilizaremos el comando rpm -hiv nombre_del_paquete.rpm

para la instalación de los paquetes.

Después de instalados los paquetes debemos seguir una serie de procedimientos mostrados a continuación.

Debemos crear una carpeta y darle permisos de acceso al usuario y grupo Ldap.

mkdir /var/lib/ldap/autenticar

chmod 700 /var/lib/ldap/autenticar

chown ldap.ldap /var/lib/ldap/autenticar.

Creamos la clave que se le asignara al usuario administrador del directorio con el siguiente comando.

slappasswd y generara una clave encriptada, la cual será utilizada cuando editemos los parámetros de configuración en el archivo /etc/openldap/sladp.conf

{SSHA}b8JF2tZ+ST+w5tWM+s42MafnJaUMaXQN.

Editamos el fichero /etc/openldap/sladp.conf (ver Anexo 2.3.1 sladp.conf).

Luego iniciamos el demonio del servidor openLdap con el comando

#service ldap start.

#chkconfig ldap on // inicia al arranque del sistema operativo.

Editamos el fichero

/usr/share/openldap/migration/migrate_common.ph

modificamos los valores de las variable

$DEFAULT_MAIL_DOMAIN, $DEFAULT_BASE

(Ver Anexo 2.3.2 migrate_common.ph).

Debemos crear el objeto base.ldif que tendrá el resto de los datos, del directorio.

/usr/share/openldap/migration/migrate_base.pl > base.ldif

Insertamos información en el directorio

ldapadd -x -W -D 'cn=Administrador, dc=sacs, dc=com' -h 192.168.15.254 -f base.ldif.

Debemos crear también los ficheros para almacenar la información de los grupos, de las cuentas incluyendo claves de acceso.

/usr/share/openldap/migration/migrate_group.pl /etc/group group.ldif

/usr/share/openldap/migration/migrate_passwd.pl /etc/passwd passwd.ldif

Ingresamos información a estos archivos.

ldapadd -x -W -D 'cn=Administrador, dc=sacs, dc=com' -h 192.168.15.254 -f group.ldif

ldapadd -x -W -D 'cn=Administrador, dc=sacs, dc=com' -h 192.168.15.254 -f passwd.ldif.

por último debemos configurar que la autenticación se la realice utilizando openLdap utilizando una herramienta como el authconfig-gtk.

Digitamos en una terminal authconfig-gtk damos enter y saldrá la ventana de configuración.

Figura 1.31 configuración de la autenticación.

1.3.4 Integración samba OpenLDAP

Debemos copiar las schemas de samba en el directorio de schemas de openLdap.

cp /usr/share/doc/samba-*/LDAP/samba.schema

/etc/openldap/schema/

Modificamos el archivo sladp.conf agregamos unas cuantas líneas.

Include	**/etc/openldap/schema/samba.schema**	
Index	**sambaSID**	**eq**
index	**sambaPrimaryGroupSID**	**eq**
index	**sambaDomainName**	**eq**
index	**default**	**sub**

También debemos configurar el archivo de clientes NSS /etc/ldap.conf (ver Anexo 2.3.2 ldap.conf cliente (NSS)).

Archivo cliente Ldap /etc/openLdap/ldap.conf (ver Anexo2.3.3 ldap.conf cliente (LDAP).

Se realizan cambios en el archivo smb.conf de samba (ver Anexo 2.2.1 smb.conf).

Ahora debemos hacerle saber a samba el password del administrador del openLdap con el siguiente comando.

smbpasswd -w password.

Archivo /etc/openldap/slapd.conf

```
#
# See slapd.conf(5) for details on configuration options.
# This file should NOT be world readable.
#
include         /etc/openldap/schema/core.schema
include         /etc/openldap/schema/cosine.schema
include         /etc/openldap/schema/inetorgperson.schema
include         /etc/openldap/schema/nis.schema
include         /etc/openldap/schema/samba.schema
# Allow LDAPv2 client connections.  This is NOT the default.
allow bind_v2

# Do not enable
# service AND a
#referral      ldap://root.openldap.org

pidfile         /var/run/openldap/slapd.pid
argsfile        /var/run/openldap/slapd.args

# Load dynamic backend modules:
# modulepath   /usr/lib/openldap
# moduleload   back_bdb.la
# moduleload   back_ldap.la
# moduleload   back_ldbm.la
# moduleload   back_passwd.la
# moduleload   back_shell.la

# The next three lines allow use of TLS for encrypting connections
using a
# dummy test certificate which you can generate by changing to
# /etc/pki/tls/certs, running "make slapd.pem", and fixing permissions
on
# slapd.pem so that the ldap user or group can read it.  Your client
software
# may balk at self-signed certificates, however.
# TLSCACertificateFile /etc/pki/tls/certs/ca-bundle.crt
# TLSCertificateFile /etc/pki/tls/certs/slapd.pem
# TLSCertificateKeyFile /etc/pki/tls/certs/slapd.pem

# Sample security restrictions
#       Require integrity protection (prevent hijacking)
#       Require 112-bit (3DES or better) encryption for updates
#       Require 63-bit encryption for simple bind
# security ssf=1 update_ssf=112 simple_bind=64

# Sample access control policy:
#       Root DSE: allow anyone to read it
#       Subschema (sub)entry DSE: allow anyone to read it
        #Other DSEs:
#               Allow self write access
#               Allow authenticated users read access
#               Allow anonymous users to authenticate
#       Directives needed to implement policy:
# access to dn.base="" by * read
# access to dn.base="cn=Subschema" by * read
# access to *
#       by self write
#       by users read
#       by anonymous auth

#
# if no access controls are present, the default policy
# allows anyone and everyone to read anything but restricts
# updates to rootdn.  (e.g., "access to * by * read")
#
# rootdn can always read and write EVERYTHING!
```

```
#######################################################################
# ldbm and/or bdb database definitions
#######################################################################

database       bdb
suffix         "dc=sacs,dc=com"
rootdn         "cn=Administrador,dc=sacs,dc=com"
# Cleartext passwords, especially for the rootdn, should
# be avoided.  See slappasswd(8) and slapd.conf(5) for details.
# Use of strong authentication encouraged.
# rootpw               secret
rootpw        {SSHA}b8JF2tZ+ST+w5tWM+s42MafnJaUMaXQN
# The database directory MUST exist prior to running slapd AND
# should only be accessible by the slapd and slap tools.
# Mode 700 recommended.
directory      /var/lib/ldap/autenticar

# Indices to maintain for this database
#index objectClass                                  eq,pres
#index ou,cn,mail,surname,givenname         eq,pres,sub
#index uidNumber,gidNumber,loginShell        eq,pres
#index uid,memberUid                        eq,pres,sub
#index nisMapName,nisMapEntry               eq,pres,sub
index objectClass                                   eq,pres
index cn                                           eq,pres,sub
index sn                                           eq,pres,sub
index uid                                    eq,pres,sub
index displayName                     eq,pres,sub
index uidNumber                                    eq
index gidNumber                            eq
index memberUID                          eq
index sambaSID                             eq
index sambaPrimaryGroupSID                 eq
index sambaDomainName                      eq
index default                            sub
# Replicas of this database
#replogfile /var/lib/ldap/openldap-master-replog
#replica host=ldap-1.example.com:389 starttls=critical
#     bindmethod=sasl saslmech=GSSAPI
#     authcId=host/ldap-master.example.com@EXAMPLE.COM
```

1.3.5 Instalación y configuración de smbldap-tools

Smbldap-tools es una herramienta que nos permite administrar la integración de samba con openLDAP

Para la instalación del servicio smbldap-tools necesitamos tener instalados los siguientes paquetes:

instalamos el RPM de http://dag.wieers.com/

release/rpmforge-release-0.3.6-1.el5.rf.i386.rpm
rpm -Uvh rpmforge-release-0.3.6-1.el5.rf.i386.rpm

luego editamos el fichero /etc/yum.repos.d/rpmforge.repo :

[rpmforge]

name = Red Hat Enterprise $releasever - RPMforge.net - dag

#baseurl = http://apt.sw.be/redhat/el5/en/$basearch/dag

mirrorlist = http://apt.sw.be/redhat/el5/en/mirrors-rpmforge

#mirrorlist = file:///etc/yum.repos.d/mirrors-rpmforge

enabled = 0

protect = 0

gpgkey = file:///etc/pki/rpm-gpg/RPM-GPG-KEY-rpmforge-dag

gpgcheck = 1

Dejamos deshabilitado el repositorio por defecto (enabled=0).

Luego de esto, debemos instalar paquetes adicionales.

perl-Compress-Zlib.i386 0:1.42-1.fc6.

perl-Convert-ASN1.noarch 0:0.20-1.1

perl-Crypt-SmbHash.noarch 0:0.12-1.2.el5.rf

perl-Digest-SHA1.i386 0:2.11-1.2.1

perl-HTML-Parser.i386 0:3.55-1.fc6

perl-HTML-Tagset.noarch 0:3.10-2.1.1

perl-IO-Socket-SSL.noarch 0:1.01-1.fc6

perl-Net-SSLeay.i386 0:1.30-4.fc6

perl-URI.noarch 0:1.35-3

perl-XML-NamespaceSupport.noarch 0:1.09-1.2.1

perl-XML-SAX.noarch 0:0.14-5

perl-ldap.noarch 0:0.33-1.el5.rf

perl-libwww-perl.noarch 0:5.805-1.1.1

smbldap-tools.noarch 0:0.9.1-1.2.el5.rf

Debemos conocer SID de nuestro servidor Samba con el comando: net getlocalsid

SID for domain LINUX is: S-1-5-21-XXXXXXXXXX-XXXXXXXXXX el cual lo utilizaremos más adelante en la configuracion.

Procedemos a configurar el acceso smbldap-tools a la BD de openLDAP en /etc/smbldap-tools/smbldap_bind.conf.

Configuración principal en /etc/smbldap-tools/smbldap.conf.

Finalmente debemos crear la estructura del directorio LDAP

Utilizamos el siguiente comando

smbldap-populate -a Administrator

En este caso no se llamara root por defecto, sino que se le cambio al parámetro -a Administrator.

Listo ya podemos utilizar los comandos del smbldap-tools de la siguiente manera:

smbldap-useradd -a -m -F "" usuariowin

smbldap-passwd usuariowin

Reiniciamos samba y openLdap:

/etc/smbldap-tools/smbldap_bind.conf

```
##############################
# Credential Configuration #
##############################
# Notes: you can specify two differents configuration if you use a
# master ldap for writing access and a slave ldap server for reading access
# By default, we will use the same DN (so it will work for standard Samba
# release)
slaveDN="cn=Administrador,dc=sacs,dc=com"
slavePw="emelec"
masterDN="cn=Administrador,dc=sacs,dc=com"
masterPw="emelec"
```

```
# $Source: $
# $Id: smbldap.conf,v 1.18 2005/05/27 14:28:47 jtournier Exp $
#
# smbldap-tools.conf : Q & D configuration file for smbldap-tools

#  This code was developped by IDEALX (http://IDEALX.org/) and
#  contributors (their names can be found in the CONTRIBUTORS file).
#
#                 Copyright (C) 2001-2002 IDEALX
#
#  This program is free software; you can redistribute it and/or
#  modify it under the terms of the GNU General Public License
#  as published by the Free Software Foundation; either version 2
#  of the License, or (at your option) any later version.
#
#  This program is distributed in the hope that it will be useful,
#  but WITHOUT ANY WARRANTY; without even the implied warranty of
#  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
#  GNU General Public License for more details.
#
#  You should have received a copy of the GNU General Public License
#  along with this program; if not, write to the Free Software
#  Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307,
#  USA.

#  Purpose :
#       . be the configuration file for all smbldap-tools scripts

##############################################################################
#                                                                            #
# General Configuration                                                      #
#                                                                            #
##############################################################################

# Put your own SID. To obtain this number do: "net getlocalsid".
```

```
# If not defined, parameter is taking from "net getlocalsid" return
SID="S-1-5-21-4233777812-4109637176-3988859424"

# Domain name the Samba server is in charged.
# If not defined, parameter is taking from smb.conf configuration file
# Ex: sambaDomain="sacs"
sambaDomain="sacs"

##############################################################################
#                                                                            #
# LDAP Configuration                                                         #
#                                                                            #
##############################################################################

# Notes: to use to dual ldap servers backend for Samba, you must patch
# Samba with the dual-head patch from IDEALX. If not using this patch
# just use the same server for slaveLDAP and masterLDAP.
# Those two servers declarations can also be used when you have
# . one master LDAP server where all writing operations must be done
# . one slave LDAP server where all reading operations must be done
#   (typically a replication directory)

# Slave LDAP server
# Ex: slaveLDAP=127.0.0.1
# If not defined, parameter is set to "127.0.0.1"
slaveLDAP="192.168.15.254"

# Slave LDAP port
# If not defined, parameter is set to "389"
slavePort="389"

# Master LDAP server: needed for write operations
# Ex: masterLDAP=127.0.0.1
# If not defined, parameter is set to "127.0.0.1"
masterLDAP="192.168.15.254"

# Master LDAP port
# If not defined, parameter is set to "389"
masterPort="389"
# Use TLS for LDAP
# If set to 1, this option will use start_tls for connection
# (you should also used the port 389)
# If not defined, parameter is set to "1"
##ldapTLS="1"
# How to verify the server's certificate (none, optional or require)
# see "man Net::LDAP" in start_tls section for more details
#verify="optional"

# CA certificate
# see "man Net::LDAP" in start_tls section for more details
##cafile="/etc/smbldap-tools/ca.pem"

# certificate to use to connect to the ldap server
# see "man Net::LDAP" in start_tls section for more details
##clientcert="/etc/smbldap-tools/smbldap-tools.pem"

# key certificate to use to connect to the ldap server
# see "man Net::LDAP" in start_tls section for more details
##clientkey="/etc/smbldap-tools/smbldap-tools.key"
# LDAP Suffix
# Ex: suffix=dc=IDEALX,dc=ORG
suffix="dc=sacs,dc=com"

# Where are stored Users
# Ex: usersdn="ou=Users,dc=IDEALX,dc=ORG"
```

```
# Warning: if 'suffix' is not set here, you must set the full dn for
usersdn
usersdn="ou=People,${suffix}"

# Where are stored Computers
# Ex: computersdn="ou=Computers,dc=IDEALX,dc=ORG"
# Warning: if 'suffix' is not set here, you must set the full dn for
computersdn
computersdn="ou=Computers,${suffix}"

# Where are stored Groups
# Ex: groupsdn="ou=Groups,dc=IDEALX,dc=ORG"
# Warning: if 'suffix' is not set here, you must set the full dn for
groupsdn
groupsdn="ou=Group,${suffix}"

# Where are stored Idmap entries (used if samba is a domain member server)
# Ex: groupsdn="ou=Idmap,dc=IDEALX,dc=ORG"
# Warning: if 'suffix' is not set here, you must set the full dn for
idmapdn
idmapdn="ou=Idmap,${suffix}"

# Where to store next uidNumber and gidNumber available for new users and
groups
# If not defined, entries are stored in sambaDomainName object.
# Ex: sambaUnixIdPooldn="sambaDomainName=${sambaDomain},${suffix}"
# Ex: sambaUnixIdPooldn="cn=NextFreeUnixId,${suffix}"
sambaUnixIdPooldn="sambaDomainName=${sacs},${suffix}"

# Default scope Used
scope="sub"

# Unix password encryption (CRYPT, MD5, SMD5, SSHA, SHA, CLEARTEXT)
hash_encrypt="SSHA"
# if hash_encrypt is set to CRYPT, you may set a salt format.
# default is "%s", but many systems will generate MD5 hashed
# passwords if you use "$1$%.8s". This parameter is optional!
crypt_salt_format="%s"
##############################################################################
#
# Unix Accounts Configuration
#
##############################################################################
# Login defs
# Default Login Shell
# Ex: userLoginShell="/bin/bash"
userLoginShell="/bin/bash"

# Home directory
# Ex: userHome="/home/%U"
userHome="/home/%U"

# Default mode used for user homeDirectory
userHomeDirectoryMode="700"

# Gecos
userGecos="System User"

# Default User (POSIX and Samba) GID
defaultUserGid="513"

# Default Computer (Samba) GID
defaultComputerGid="515"

# Skel dir
skeletonDir="/etc/skel"
```

```
# Default password validation time (time in days) Comment the next line if
# you don't want password to be enable for defaultMaxPasswordAge days (be
# careful to the sambaPwdMustChange attribute's value)
defaultMaxPasswordAge="9999"

#########################################################################
#
# SAMBA Configuration
#
#########################################################################

# The UNC path to home drives location (%U username substitution)
# Just set it to a null string if you want to use the smb.conf 'logon home'
# directive and/or disable roaming profiles
# Ex: userSmbHome="\\PDC-SMB3\%U"
userSmbHome="\\server\%U"

# The UNC path to profiles locations (%U username substitution)
# Just set it to a null string if you want to use the smb.conf 'logon path'
# directive and/or disable roaming profiles
# Ex: userProfile="\\server\profiles\%U"
userProfile="\\server\profiles\%U"

# The default Home Drive Letter mapping
# (will be automatically mapped at logon time if home directory exist)
# Ex: userHomeDrive="H:"
userHomeDrive="H:"

# The default user netlogon script name (%U username substitution)
# if not used, will be automatically username.cmd
# make sure script file is edited under dos
# Ex: userScript="startup.cmd" # make sure script file is edited under dos
userScript="logon.cmd"

# Domain appended to the users "mail"-attribute
# when smbldap-useradd -M is used
# Ex: mailDomain="idealx.com"
mailDomain="sacs.com"

#########################################################################
#
# SMBLDAP-TOOLS Configuration (default are ok for a RedHat)
#
#########################################################################

# Allows not to use smbpasswd (if with_smbpasswd == 0 in smbldap_conf.pm)
but
# prefer Crypt::SmbHash library
with_smbpasswd="0"
smbpasswd="/usr/bin/smbpasswd"

# Allows not to use slappasswd (if with_slappasswd == 0 in smbldap_conf.pm)
# but prefer Crypt:: libraries
with_slappasswd="0"
slappasswd="/usr/sbin/slappasswd"

# comment out the following line to get rid of the default banner
# no_banner="1"
```

1.3.6 Configuración de la Base de Datos

Los paquetes para poder utilizar la base de datos son los siguientes:

mysql-5.0.22-2.1.i386.rpm

mysql-server-5.0.22-2.1.i386.rpm

Luego de instalar los paquetes, ejecutamos los siguientes comandos:

/sbin/service mysqld start

/sbin/chkconfig –level 345 mysqld on

1.3.7 Configuración del VirtualHost

Esta configuración la hacemos por medio del protocolo Httpd, en la cual procedemos a configurar esta seccion del archivo httpd.conf, mediante la ruta /etc/httpd/conf/httpd.conf

<VirtualHost *:80>

```
    ServerAdmin root@sacs.com
    DocumentRoot /var/www/sacs
    ServerName www.sacs.com
    ProxyRequests Off
    ProxyPassReverse /SACS http://www.sacs.com:8888/SACS
    ProxyPass /SACS http://www.sacs.com:8888/SACS
    ProxyPreserveHost On
    Redirect /index.jsp http://www.sacs.com/SACS
#   Redirect http://www.sacs.com http://www.sacs.com
#   ErrorLog logs/dummy-host.example.com-error_log
#   CustomLog logs/dummy-host.example.com-access_log common
</VirtualHost>
```

1.3.8 Configuración de VSFTP (Very Secure File Transfer Protocol)

El vsftp es un protoco9lo para transferir archivos utilizando TCP/IP, es muy utilizado en Internet en servidores de descargas de archivos y utiliza el puerto 20 para flujo de datos y el puerto 21 para órdenes.

Paquetes que se necesitan:

vsftpd-2.0.5-10.el5.i386.rpm

Se encuentra en la siguiente ruta:

/etc/vsftpd/vsftpd.conf

Archivos de configuración:

/etc/vsftpd/vsftpd.conf : Fichero de configuracion

/etc/vsftpd.user_list : Lista que definirá usuarios a enjaular o no a enjaular, dependiendo de la configuración.

Utilizamos un editor de texto y modificamos el fichero **/etc/vsftpd/vsftpd.conf**. A continuación analizaremos los parámetros a modificar o añadir, según se requiera para nuestras necesidades particulares.

Parámetros de vsftpd.conf

Anonymous-enable

Se utiliza para definir si se permitirán los accesos anónimos al servidor. Y se establece como valor **YES** o **NO** de acuerdo a lo que se requiera.

anonymous-enable = YES

local_enable

Establece si se van a permitir los accesos autenticados de los usuarios locales del sistema.

local_enable = YES

write_enable

Establece si se permite el mandato write (escritura) en el servidor.

write_enable = YES

ftpd_banner

Este parámetro sirve para establecer el banderín de bienvenida que será mostrado cada vez que un usuario acceda al servidor.

ftpd_banner=Bienvenido al servidor FTP del Servidor de Archivo Corporativo

chroot_local_user

Cuando los usuarios del sistema se autentiquen tendrán acceso a otros directorios del sistema fuera de su directorio personal. Para recluir a los usuarios a solo poder utilizar su propio directorio personal, utilizaremos el parámetro **chroot_local_user** que habilitará la función de **chroot()** y los parámetros chroot_list_enable y chroot_list_file para establecer el fichero con la lista de usuarios que quedarán excluidos de la función **chroot()**.

chroot_local_user=YES

chroot_list_enable=YES

chroot_list_file=/etc/vsftpd/vsftpd.chroot_list

Despues de este paso, cada vez que un usuario local se autentique en el servidor FTP, solo tendrá acceso a su propio directorio personal y lo que este contenga

Controlar el ancho de banda utilizaremos los siguientes parámetros:

anon_max_rate

Se utiliza para limitar la tasa de transferencia en bytes por segundo para los usuarios anónimos.

anon_max_rate = 5120

max_clients

Establece el número máximo de clientes que podrán acceder simultáneamente hacia el servidor FTP.

max_clients = 5

max_per_ip

Establece el número máximo de conexiones que se pueden realizar desde una misma dirección IP.

max_per-ip = 5

Para iniciar, reiniciar o detener el servicio vsftpf

Para iniciar por primera vez el servicio, debemos utilizar:

service vsftpd start

Para hacer que los cambios hechos a la configuración se actualicen:

service vsftpd restart

Para detener el servicio:

```
# Example config file /etc/vsftpd/vsftpd.conf
#
# The default compiled in settings are fairly paranoid. This sample file
# loosens things up a bit, to make the ftp daemon more usable.
# Please see vsftpd.conf.5 for all compiled in defaults.
#
# READ THIS: This example file is NOT an exhaustive list of vsftpd options.
# Please read the vsftpd.conf.5 manual page to get a full idea of vsftpd's
# capabilities.
#
# Allow anonymous FTP? (Beware - allowed by default if you comment this out).
anonymous_enable=NO
#
# Uncomment this to allow local users to log in.
local_enable=YES
#
# Uncomment this to enable any form of FTP write command.
write_enable=YES
#
# Default umask for local users is 077. You may wish to change this to 022,
# if your users expect that (022 is used by most other ftpd's)
local_umask=022
#
# Uncomment this to allow the anonymous FTP user to upload files. This only
# has an effect if the above global write enable is activated. Also, you will
# obviously need to create a directory writable by the FTP user.
#anon_upload_enable=YES
#
# Uncomment this if you want the anonymous FTP user to be able to create
# new directories.
#anon_mkdir_write_enable=YES
#
# Activate directory messages - messages given to remote users when they
# go into a certain directory.
dirmessage_enable=YES
#

# Activate logging of uploads/downloads.
xferlog_enable=YES
#
# Make sure PORT transfer connections originate from port 20 (ftp-data).
connect_from_port_20=YES
#
# If you want, you can arrange for uploaded anonymous files to be owned by
# a different user. Note! Using "root" for uploaded files is not
# recommended!
#chown_uploads=YES
#chown_username=whoever
#
# You may override where the log file goes if you like. The default is shown
# below.
#xferlog_file=/var/log/vsftpd.log
#
# If you want, you can have your log file in standard ftpd xferlog format
xferlog_std_format=YES
#
# You may change the default value for timing out an idle session.
#idle_session_timeout=600
#
# You may change the default value for timing out a data connection.
#data_connection_timeout=120
#
# It is recommended that you define on your system a unique user which the
# ftp server can use as a totally isolated and unprivileged user.
#nopriv_user=ftpsecure
#
# Enable this and the server will recognise asynchronous ABOR requests. Not
# recommended for security (the code is non-trivial). Not enabling it,
```

```
# however, may confuse older FTP clients.
#async_abor_enable=YES
#
# By default the server will pretend to allow ASCII mode but in fact ignore
# the request. Turn on the below options to have the server actually do ASCII
# mangling on files when in ASCII mode.
# Beware that on some FTP servers, ASCII support allows a denial of service
# attack (DoS) via the command "SIZE /big/file" in ASCII mode. vsftpd
# predicted this attack and has always been safe, reporting the size of the
# raw file.
# ASCII mangling is a horrible feature of the protocol.
#ascii_upload_enable=YES
#ascii_download_enable=YES
#
# You may fully customise the login banner string:
ftpd_banner=Bienvenidos a www.sacs.com.
#
# You may specify a file of disallowed anonymous e-mail addresses. Apparently
# useful for 55ombating certain DoS attacks.
#deny_email_enable=YES
# (default follows)
#banned_email_file=/etc/vsftpd/banned_emails
#
# You may specify an explicit list of local users to chroot() to their home
# directory. If chroot_local_user is YES, then this list becomes a list of
# users to NOT chroot().
chroot_local_user=YES
chroot_list_enable=YES
# (default follows)
chroot_list_file=/etc/vsftpd/chroot_list
#
# You may activate the "-R" option to the builtin ls. This is disabled by
# default to avoid remote users being able to cause excessive I/O on large
# sites. However, some broken FTP clients such as "ncftp" and "mirror" assume
# the presence of the "-R" option, so there is a strong case for enabling it.
#ls_recurse_enable=YES
#
# When "listen" directive is enabled, vsftpd runs in standalone mode and
# listens on Ipv4 sockets. This directive cannot be used in conjunction
# with the listen_ipv6 directive.
listen=YES
#
# This directive enables listening on Ipv6 sockets. To listen on Ipv4 and Ipv6
# sockets, you must run two copies of vsftpd whith two configuration files.
# Make sure, that one of the listen options is commented !!
#listen_ipv6=YES
pam_service_name=vsftpd
userlist_enable=YES
tcp_wrappers=YES
anon_max_rate=5120
local_max_rate=5120
max_clients=5
max_per_ip=5
# Example file /etc/vsftpd.user_list
#

# vsftpd userlist
# If userlist_deny=NO, only allow users in this file
# If userlist_deny=YES (default), never allow users in this file, and
# do not even prompt for a password.
# Note that the default vsftpd pam config also checks /etc/vsftpd/ftpusers
# for users that are denied.
root
bin
daemon
adm
lp
```

1.3.8 Configuración de Sendmail

Sendmail es el más popular agente de transporte de correo (MTA o **M**ail **T**ransport **A**gent), electrónico del mundo.

Para configuración sendmail vamos a necesitar los siguientes paquetes:

• sendmail

• make

• sendmail-cf

• cyrus-sasl

• dovecot (o bien imap)

• cyrus-sasl-md5

• m4 • cyrus-sasl-plain

Dovecot es un servidor de POP3 e IMAP de fuente abierta que funciona en Linux y sistemas basados sobre Unix™ y está diseñado con la seguridad como principal objetivo.

Los protocolos que se utilizan para esta configuración son SMTP, POP3 e IMAP.

SMTP es un protocolo estándar de Internet del Nivel de Aplicación utilizado para la transmisión de correo electrónico a través de una conexión TCP/IP, trabaja sobre TCP en el puerto 25.
POP3 es un protocolo estándar de Internet del Nivel de Aplicación que recupera el correo electrónico desde un servidor remoto a través de una conexión TCP/IP desde un cliente local, trabaja sobre TCP en el puerto 110.

IMAP es un protocolo estándar de Internet del Nivel de Aplicación utilizado para acceder hacia el correo electrónico en un servidor remoto a través de una conexión TCP/IP desde un cliente local, trabaja sobre TCP en el puerto 143.

Para determinar el servidor SMTP para un dominio dado, se utilizan los registros MX (Mail Exchanger) en la Zona de Autoridad correspondiente a ese mismo dominio contestado por un Servidor DNS. Después de establecerse una conexión entre el remitente (el cliente) y el destinatario (el servidor), se inicia una sesión SMTP, ejemplificada a continuación.

Cliente: $ telnet 127.0.0.1 25
Servidor: Trying 127.0.0.1...

```
                Connected to localhost.localdomain (127.0.0.1).
                Escape character is '^]'.
                220 nombre.dominio ESMTP Sendmail 8.13.1/8.13.1; Sat, 18 Mar
                2006
                16:02:27
                -0600
Cliente:    HELO localhost.localdomain
Servidor: 250 nombre.dominio Hello localhost.localdomain [127.0.0.1],
                pleased to
                meet you
Cliente:    MAIL FROM:<fulano@localhost.localdomain>
Servidor:  250 2.1.0 <fulano@localhost.localdomain>... Sender ok
Cliente:    RCPT TO:<root@localhost.localdomain>
Servidor: 250 2.1.5 <root@localhost.localdomain>... Recipient ok
Cliente:    ATA
Servidor: 354 Enter mail, end with "." on a line by itself
Cliente:    Subject: Mensaje de prueba
                From:  fulano@localhost.localdomain
                To: root@localhost.localdomain
                Hola. Este es un mensaje de prueba.
                Adios.
.
Servidor: 250 2.0.0 k2IM2RjA003987 Message accepted for delivery
Cliente:    QUIT
Servidor: 221 2.0.0 nombre.dominio closing connection
Servidor: Connection closed by foreign host.
```

```
          for <root@localhost.localdomain>; Sat, 18 Mar 2006 16:03:21
          -0600
          Date: Sat, 18 Mar 2006 16:02:27 -0600
          Message-Id:
          <200603182203.k2IM2RjA003987@localhost.localdomain>
          Subject: Mensaje de prueba
          From: fulano@localhost.localdomain
          To: root@localhost.localdomain
          Status: O
          Content-Length: 43
          Lines: 2
          X-UID: 202
          X-Keywords:
          Hola. Este es un mensaje de prueba.
```

Adios.

.
Cliente: QUIT
Servidor:+OK Logging out.
Connection closed by foreign host.

1.4 Codificación.

1.4.1 Clases Principales

1.4.1.1 Grupo.java

En esta clase definimos las funciones para setear y capturar los atributos de la tabla grupos, además contiene funciones para manipular la Base de Datos como insertar, consultar y actualizar.

```
package fileserver.clases;

import fileserver.Conexion;
import fileserver.Consultas;
import fileserver.insertar;

public class Grupo
{

  private Long idGrupo;
  private String descripcion;
  private String estado;
  private Double cuota;


  public Grupo()
  {
  }
  public Long getIdGrupo()
  {
    return idGrupo;
  }

  public void setIdGrupo(Long idGrupo)
  {
    this.idGrupo = idGrupo;
  }

  public String getDescripcion()
  {
    return descripcion;
  }
```

```
  public void setDescripcion(String descripcion)
  {
    this.descripcion = descripcion;
  }

  public String getEstado()
  {
    return estado;
  }

  public void setEstado(String estado)
  {
    this.estado = estado;
  }


  public Object ObtenerGrupo(Conexion apli)
   {

     Grupo c = new Grupo();

     try
     {
      c=(Grupo)Consultas.Informacion(Grupo.class, "SELECT
id_grupo,descripcion,estado FROM fil_grupo where id_grupo=" +
getIdGrupo() , apli);
       setIdGrupo(c.getIdGrupo());
       setDescripcion(c.getDescripcion());
       setEstado(c.getEstado());
     }
     catch (Exception ex)
     {
         ex.printStackTrace();
     }

     return c;
   }
  public Object ObtenerIdGrupo(Conexion apli)
   {

     Grupo c = new Grupo();

     try
     {
      c=(Grupo)Consultas.Informacion(Grupo.class, "SELECT
id_grupo,descripcion,estado FROM fil_grupo where descripcion like '" +
getDescripcion() + "'" , apli);
       setIdGrupo(c.getIdGrupo());
       setDescripcion(c.getDescripcion());
       setEstado(c.getEstado());
     }

     catch (Exception ex)
     {
         ex.printStackTrace();
     }

     return c;
   }
```

```
 public boolean Modificar(Conexion conex)
   { boolean ret=false;
     try
     {
       ret=insertar.Guardar("UPDATE fil_grupo set
estado='"+getEstado()+"' where id_grupo=" + getIdGrupo() ,conex);
     }
     catch (Exception ex)
     { ex.printStackTrace(); }
     return ret;
   }
  public java.util.Vector ObtenerArregloGrupo(Conexion conex)
   {

     java.util.Vector c=new java.util.Vector();

     try
     {
      c=Consultas.InformacionArreglo(Grupo.class, "SELECT
id_grupo,descripcion,estado FROM fil_grupo", conex);
     }
     catch (Exception ex)
     {
         ex.printStackTrace();
     }

     return c;
   }
  public Double getCuota()
  {
    return cuota;
  }

  public void setCuota(Double cuota)
  {
    this.cuota = cuota;
  }

  public boolean ModificarCuota(Conexion conex)
   { boolean ret=false;
     try
     {
       ret=insertar.Guardar("UPDATE fil_grupo set cuota="+getCuota()+"
where id_grupo='" + getIdGrupo() + "'",conex);
     }
     catch (Exception ex)
     { ex.printStackTrace(); }
     return ret;
   }
     public java.util.Vector ObtenerArregloGrupoActivos(Conexion
conex)
   {

     java.util.Vector c=new java.util.Vector();

     try
     {
      c=Consultas.InformacionArreglo(Grupo.class, "SELECT
id_grupo,descripcion,estado FROM fil_grupo where fil_grupo.estado=1",
conex);
     }
     catch (Exception ex)
     {
         ex.printStackTrace();
     }

     return c;
   }
}
```

1.4.1.2 Usuario.java

En esta clase definimos las funciones para setear y capturar los atributos de la tabla usuarios, además contiene funciones para manipular la Base de Datos

```
package fileserver.clases;

import fileserver.Conexion;
import fileserver.Consultas;

import java.sql.SQLException;

public class Usuario
{

  private String idUsuario;
  private String nombre;
  private String clave;
  private String estado;

  public Usuario()
  {
  }

  public String getIdUsuario()
  {
    return idUsuario;
  }

  public void setIdUsuario(String idUsuario)
  {
    this.idUsuario = idUsuario;
  }

  public String getNombre()
  {
    return nombre;
  }
  public void setNombre(String nombre)
  {
    this.nombre = nombre;
  }

  public String getClave()
  {
    return clave;
  }

  public void setClave(String clave)
  {
    this.clave = clave;
  }
```

```
  public String getEstado()
  {
    return estado;
  }
  public void setEstado(String estado)
  {
    this.estado = estado;
  }

 public String prueba(Conexion apli)
   {

     Usuario c = new Usuario();

     try
     {
       c=(Usuario)Consultas.Informacion(Usuario.class, "SELECT
id_usuario FROM fil_usuario", apli);
     }
     catch (Exception ex)
     {
         ex.printStackTrace();
     }

     return c.idUsuario;
   }

  public static void main (String agrs[])
    throws SQLException
  {
  Usuario unUsuario =new Usuario();
  Conexion co=new Conexion();
  System.out.println( unUsuario.prueba(co));

  }


}
```

1.4.1.3 Perfil.java

```
package fileserver.clases;

public class Perfil
{
  public Perfil()
  {
  }
}
```

1.4.1.4 UsuarioPerfil.java

La clase Usuarioperfil maneja las funciones de seteo y captura de los atributos de la tabla usuarioperfil, tanto para el usuario como para el perfil, también se realiza las funciones de inserción a la base de datos de la relación

```
package fileserver.clases;

import fileserver.Conexion;
import fileserver.Consultas;
import fileserver.insertar;

public class UsuarioPerfil
{
  private String idUsuario;
  private Long idPerfil;
  private Usuario unUsuario;
  private Perfil unPerfil;

  public UsuarioPerfil()
  {
  }
  public String getIdUsuario()
  {
    return idUsuario;
  }
```

```
  public void setIdUsuario(String idUsuario)
  {
    this.idUsuario = idUsuario;
  }

  public Long getIdPerfil()
  {
    return idPerfil;
  }

  public void setIdPerfil(Long idPerfil)
  {
    this.idPerfil = idPerfil;
  }

  public Usuario getUnUsuario()
  {
    return unUsuario;
  }
  public void setUnUsuario(Usuario unUsuario)
  {
    this.unUsuario = unUsuario;
  }

  public Perfil getUnPerfil()
  {
    return unPerfil;
  }

  public void setUnPerfil(Perfil unPerfil)
  {
    this.unPerfil = unPerfil;
  }
public void ObtenerLogon(Conexion conexion,String IdLogon)
   {
     try
     {
       unUsuario=new Usuario();
       unUsuario.setIdUsuario(IdLogon);
       unUsuario.ObtenerUsuario(conexion);
       if(unUsuario!=null || unUsuario.getIdUsuario()!=null)
       {
         unPerfil=new Perfil();
         unPerfil.ObtenerPerfil(conexion,IdLogon);
         setIdUsuario(unUsuario.getIdUsuario());
         setIdPerfil(unPerfil.getIdPerfil());
       }
     }
     catch (Exception ex)
     {
         ex.printStackTrace();
     }

   }

  public boolean Guardar(Conexion conex)
   { boolean ret=false;
     try
     {
```

```
       ret=insertar.Guardar("INSERT INTO fil_usuario_perfil
(id_usuario,id_perfil) VALUES ('"+getIdUsuario()
+"','"+getIdPerfil()+")",conex);
     }
     catch (Exception ex)
     { ex.printStackTrace(); }
     return ret;
   }
}
```

1.4.1.5 jftp.java

La clase jftp incluye las funciones necesarias para subir (putFTP), descargas archivos (getFTP), crear archivos (crearArchivoFTP), crear carpetas (crearCarpetasFTP), listar archivos y carpetas (listaFTP) desde y hacia el

```
package fileserver.clases;

import  sun.net.ftp.FtpClient;
import  sun.net.TelnetOutputStream;
import com.enterprisedt.net.ftp.FTPFile;
import com.enterprisedt.net.ftp.FileTransferClient;
import java.io.IOException;

import com.enterprisedt.net.ftp.FTPException;

import java.text.ParseException;
import sun.security.krb5.internal.crypto.c;
public class jftp
{
  FileTransferClient ftp = new FileTransferClient();

  public jftp(String servidor,String usuario,String clave)
    throws IOException, FTPException
  {

    ftp.setRemoteHost(servidor);
    ftp.setUserName(usuario);
    ftp.setPassword(clave);

  }
  public void getFTP(String archivo,String dir)
   throws IOException, FTPException
  {
  ftp.connect();
  ftp.changeDirectory(dir);
  ftp.downloadFile("c:\\" + archivo,archivo);
  }
```

```
   public void putFTP(String archivo,String dir)
    throws IOException, FTPException
  {
    ftp.connect();
    ftp.changeDirectory(dir);
    ftp.uploadFile("c:\\" + archivo,archivo);
  }
  public String[] listaFTP(String dir)
  {
    String[] files=null;
    try{
    ftp.connect();
    ftp.changeDirectory(dir);
    //FTPFile[] files = ftp.directoryList(".");
    files=ftp.directoryNameList();
    ftp.disconnect();

    }
    catch (Exception ex)
    {
      ex.printStackTrace();
    }
    return files;
  }
  public void cambiarDirectorioFTP(String dir)
   throws IOException, FTPException, ParseException
  {
    ftp.connect();
    ftp.changeDirectory(dir);
  }


  public void crearCarpetaFTP(String carpeta,String dir)
   throws IOException, FTPException, ParseException
  {
    ftp.connect();
    ftp.changeDirectory(dir);
    ftp.createDirectory(carpeta);
  }


public void crearArchivoFTP(String carpeta,String dir,String
host,String nomb,String clav)
   throws IOException, FTPException, ParseException
  {
    String servidor = host;
    String userName=nomb;
    String pw=clav;
    FtpClient ftpClient = new FtpClient ();
    ftpClient.openServer (servidor);
    ftpClient.login(userName, pw);
    ftpClient.cd(dir);
    //ftpClient.binary(); ftpClient.binary ();
    TelnetOutputStream netOut = ftpClient.put(carpeta);
    netOut.close();
    ftpClient.closeServer();
  }

}
```

1.4.1.6 Conexión.java

La clase Conexión incluye las funciones necesarias para conectar y desconectar la aplicación hacia la base de datos.

Función de commit y rollback para grabar las

```
package fileserver;

import java.sql.Connection;
import java.sql.SQLException;

public class Conexion{
    private Connection connection;
    private String idUsuarioSistema;

    public Conexion()throws SQLException
  {
      java.sql.DriverManager.registerDriver(new
com.mysql.jdbc.Driver());
    }

    public Connection getConnection() throws SQLException {
        if(connection==null || connection.isClosed())
        {

connection=java.sql.DriverManager.getConnection("jdbc:mysql://db:3306/fi
leserver", "fileserver", "fileserver");
            connection.setAutoCommit(false);
        }
        return connection;
    }

    public String getIdUsuarioSistema() {
        return idUsuarioSistema;
    }

    public void setIdUsuarioSistema(String idUsuarioSistema) {
        this.idUsuarioSistema = idUsuarioSistema;
    }
  public String commit()
    {
      String err=null;
      try
      {
        Connection c=this.getConnection();
        c.commit();
```

```
      }
      catch (Exception ex)
      {
        err=ex.toString();
        ex.printStackTrace();
      }
      return err;
    }


 public String rollback()
  {
    String err=null;
    try
    {
      Connection c=this.getConnection();
      c.rollback();
    }
    catch (Exception ex)
    {
      err=ex.toString();
      ex.printStackTrace();
    }
    return err;
  }

}
```

1.4.2 J

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>

<jsp:useBean id="usuario" class="fileserver.clases.Usuario"
scope="session"/>


<html>
<head>
 <title>Abrir Archivo</title>
 <meta http-equiv="Content-Type" content="text/html;
  charset=iso-8859-1">
 <link type="text/css" rel="stylesheet" href="css/estilo.css"/>
 <LINK HREF ="css/stylesheet.css" REl="stylesheet"
  TYPE="text/css"  media="all" />
</head>

<BODY leftmargin=0 topmargin=0 marginheight="0" marginwidth="0"
bgcolor="#ffffff">

<table width="100%" heigth="100%" border="0" cellpadding="0"
 cellspacing="0"><tr> <td>
<jsp:include flush="true" page="jcabecera.jsp"/>
</td> </tr> <tr> <td>
<table width="100%" heigth="100%" border="0" cellpadding="0"
  cellspacing="0">
<tr>
<td width="50%" background="images/bg.gif" valign="top"><img
  src="images/px1.gif" width="1" height="1" alt=""
  border="0"></td>
```

```
<td valign="top" background="images/bg_left.gif"><img
  src="images/bg_left.gif" alt="" width="17" height="16"
  border="0"></td>
  <td align="left" valign="top">
   <%--tabla2--%>
<table border="0" width="681"  cellpadding="0"cellspacing="0" >
 <tr> <td width="8" align="left" valign="top"
class="body_left_bg"> </td> <td align="left" valign="top"
class="body_text_box">
  <%--inicio tabla 3--%>
<table border="0" width="130" align="left" cellpadding="0"
cellspacing="0"> <tr> <TD></TD> </tr> <tr> <td align="left"
valign="top">
 <jsp:include flush="true" page="jmenuIzquierdo.jsp"/>
 </td> </tr> </table>
 <%--fin tabla 3--%>
 </td>
 <td align="left" valign="top" class="body_text_box">
 <%--inicio tabla 4--%>
<%-------------------------------------------------------------%>

<jsp:useBean id="conexion" class="fileserver.Conexion"
 scope="session"/>
 <c:set var="directorio" value="${param['directorio']=='.'}"/>
  <c:set var="ba" value="${param['ba']==0}"/>
   <%
    java.util.Vector directory= new java.util.Vector();
    String server = "192.168.15.254";
    String user =
    (String)session.getAttribute("logon_idusuario");
    String passwd =
    (String)session.getAttribute("logon_clave");
    fileserver.clases.jftp ftpp = new
    fileserver.clases.jftp(server,user,passwd);
     String es="";
     String direccion="";
     Integer chequeo=null;
      if (request.getParameter("directorio")==null){
       directory.add("/home/" + user);
       direccion="/home/" + user;
       chequeo= 1;
       }else{

System.out.println((Integer)session.getAttribute("chequeo"));

System.out.println(Integer.parseInt(request.getParameter("estad
o")));
if (Integer.parseInt(request.getParameter("estado"))!
=(Integer)session.getAttribute("chequeo")){
chequeo =Integer.parseInt(request.getParameter("estado"));
directory=(java.util.Vector) session.getAttribute("directory");
if
(Integer.parseInt(request.getParameter("estado"))>(Integer)sess
ion.getAttribute("chequeo")){
directory.add("/" + request.getParameter("directorio"));
 }else{
  directory.removeElementAt(directory.size()-1);
 }
 for(int i=0;i<directory.size();i++)
 {
  direccion=direccion + directory.elementAt(i).toString();
 }
}
 else{
  chequeo =Integer.parseInt(request.getParameter("estado"));
  directory=(java.util.Vector)
  session.getAttribute("directory");
   //directory.add("/" + request.getParameter("directorio"));
   for(int i=0;i<directory.size();i++)
   {direccion=direccion + directory.elementAt(i).toString();}
```

```
  }
 }
 String[] cadena=null;

 System.out.println(direccion);

 cadena=ftpp.listaFTP(direccion);

 java.util.Vector grupos= new java.util.Vector();
 java.util.Vector archivos= new java.util.Vector();


 for (int i = 0; i < cadena.length; i++) {
  es=cadena[i].toString();
  //System.out.println(es.indexOf(0));
  boolean band=false;
  for (int j = 0; j < es.length(); j++) {
   if (es.charAt(j)=='.')
       band=true;
   }
   if (band==false){
     grupos.add(cadena[i].toString());
   }
   if (band==true){
     archivos.add(cadena[i].toString());
   }
   }
   pageContext.setAttribute("grupos", grupos);
   pageContext.setAttribute("archivos", archivos);
   session.setAttribute("direccion", direccion);
   session.setAttribute("chequeo", chequeo);
   session.setAttribute("directory", directory);

fileserver.clases.jftp("192.168.15.254","jreyes","jreyes123");
 %>

  <script type="text/javascript">
   function abrir(archiv)
 {

 var miapp=document.getElementById("miapplet");

 miapp.addComando("ver," + archiv);//txtarchivo.value);
 alert(archiv);
 }
 </script>

<table border="1" width="350" cellpadding="0" cellspacing="0">

 <tr>
 <td><b>DIRECTORIO.....
 </b></td>
 </tr>
 <tr>
 <td><b>.</b></td>
 </tr>
 <c:if test="${chequeo>1}">
 <c:if test="${chequeo<=2}">
  <tr>
  <td>
  <a href="jabrirArchivo.jsp"><-</a>
  </td>
  </tr>
  </c:if>
  <c:if test="${chequeo>2}">
  <tr>
  <td>
  <a href="jabrirArchivo.jsp?directorio=${grupos}&estado=$
{chequeo-1}"><-</a>
  </td>
  </tr>
```

```
  </c:if>
 </c:if>
 <c:forEach var="grupos" items="${grupos}" >
  <tr>
   <td>
    <a href="jabrirArchivo.jsp?directorio=${grupos}&estado=$
{chequeo+1}">${grupos}/</a>
  </td>
   </tr>
   </c:forEach>

   <c:forEach var="archivo" items="${archivos}" >
   <%
   request.removeAttribute("isocupado");

  String archivoc =(String)pageContext.getAttribute("archivo");
  fileserver.clases.Ocupado ocup =new
   fileserver.clases.Ocupado();
  ocup.setArchivo(archivoc);
  ocup.ObtenerArchivoOcupado(conexion);
  boolean existeGrupo=!(ocup.getArchivo()==null ||
  ocup.getArchivo().equals(""));
  if(existeGrupo)
    {
     request.setAttribute("isocupado", new Boolean(true));
    }

   %>
    <c:if test="${isocupado}">
     <tr>
      <td>
       ${archivo} (ocupado)
      </td>
       </tr>
       </c:if>
      <c:if test="${!isocupado}">
       <tr>
       <td>
                    <%--a href="jaccionAbrirArchivo.jsp?
directorio=${direccion}&archivoc=${archivo}&estado=${chequeo}"
>${archivo}< / a--%>
<a href="javascript:void(0)" onclick="bajar('${direccion}','$
{archivo}','${chequeo}')" >${archivo}</a>
       </td>
       </tr>
      </c:if>
     </c:forEach>

     </table>
      <script type="text/javascript">
      function bajar(direccion, archivo, chequeo)
      {
       var url= "jbajarArchivoHTTP.jsp?
directorio="+escape(direccion)+
                    "&archivoc="+escape(archivo)
+"&estado="+chequeo;
       var myapplet = document.getElementById("myapplet");
       myapplet.addComando("bajar", url);
       setTimeout("refrescar()", 2000);
      }
      function refrescar()
      {
       var myapplet = document.getElementById("myapplet");
       if(myapplet.getIsDone())
       {
        window.location.reload();
       }
       else setTimeout("refrescar()", 5000);
       }
```

```
        }
 </script>
 <jsp:plugin archive="GeneraArchivo.jar"
  code="fileserver.GeneraArchivo" type="applet">
  </jsp:plugin>
  <applet id="myapplet" archive="GeneraArchivo.jar"
   code="fileserver.GeneraArchivo" width="10" height="10">
 </applet>


<%-------------------------------------------------------------%>
<%--fin tabla 4--%>
</td>
<td width="8" align="left" valign="top"
class="body_right_bg"> </td>
</tr>
</table>
<%--fin tabla 2--%>
</td> <td valign="bottom" background="images/bg_right.gif"><img
src="images/bg_right.gif" alt="" width="17" height="16"
border="0"></td>
<td width="50%" background="images/bg.gif" valign="top"><img
src="images/px1.gif" width="1" height="1" alt=""
border="0"></td> </tr>
</table>

</td></tr> <tr> <td>
       <jsp:include flush="true" page="jpie.jsp"/>
</td> </tr> </table>

</BODY>
</html>
```

1.4.2.2 jaccionAbrirArchivo.jsp

El jsp jaccionAbrirArchivo contiene las instrucciones necesarias para poder abrir un archivo desde el equipo cliente a través de la aplicación.

```
<jsp:useBean id="conexion" class="fileserver.Conexion"
 scope="session"/>
 <c:set var="directorio" value="${param['directorio']=='.'}"/>
 <c:set var="ba" value="${param['ba']==0}"/>
```

```
String archivoc=request.getParameter("archivoc");
String direccion="";
Integer chequeo=null;
if (request.getParameter("directorio")==null){
 directory.add("/home/" + user);
 direccion="/home/" + user;
 chequeo= 1;
 }else{

System.out.println((Integer)session.getAttribute("chequeo"));

System.out.println(Integer.parseInt(request.getParameter("estad
o")));
if (Integer.parseInt(request.getParameter("estado"))!
=(Integer)session.getAttribute("chequeo")){
 chequeo =Integer.parseInt(request.getParameter("estado"));
 directory=(java.util.Vector)
session.getAttribute("directory");
                if
(Integer.parseInt(request.getParameter("estado"))>(Integer)sess
ion.getAttribute("chequeo")){
directory.add("/" + request.getParameter("directorio"));
 }else{
 directory.removeElementAt(directory.size()-1);
 }

 for(int i=0;i<directory.size();i++)
 {
  direccion=direccion + directory.elementAt(i).toString();
 }
}
else{
 chequeo =Integer.parseInt(request.getParameter("estado"));
 directory=(java.util.Vector)
session.getAttribute("directory");
for(int i=0;i<directory.size();i++)
{
 direccion=direccion + directory.elementAt(i).toString();
 }
}
}
System.out.println(server);
System.out.println(user);
System.out.println(passwd);
System.out.println(archivoc);
System.out.println(direccion);

fileserver.clases.Ocupado ocup =new
fileserver.clases.Ocupado();
 ocup.setArchivo(archivoc);
 ocup.ObtenerArchivoOcupado(conexion);
 boolean existeGrupo=!(ocup.getArchivo()==null ||
ocup.getArchivo().equals(""));
if(!existeGrupo)
{
 ocup.setArchivo(archivoc);
 ocup.setIdUsuario(user);
 ocup.setDireccion(direccion);
 ocup.GuardarHistorial(conexion);
 ocup.GuardarOcupado(conexion);
}
else
{
 error="Archivo esta Ocupado";
 }
 if(error==null)
 {
  error="Archivo Abierto....";
  conexion.commit();
 }
  else conexion.rollback();
      pageContext.setAttribute("error", error);
```

```
 pageContext.setAttribute("error", error);

 System.out.println("applet");
 %>
<table border="1" width="350" cellpadding="0" cellspacing="0">

<tr>
<td><b>DIRECTORIO.....
</b></td>
</tr>
<tr>
<td>
${error}
</td>
<td>
<a href="jabrirArchivo.jsp">Volver</a>
</td>
 </tr>

</table>
```

1.4.2.3 jaccionAsignarCuotaG.jsp

El jsp jaccionAsignarCuotaG permite ejecutar la acción de asignar un espacio en el Disco Duro exclusivo para un grupo de usuarios en particular a través de la

```
<c:import url="jactualizarDatos.jsp"/>
<%

 String error=null;
 //Long cuotas= request.getParameter("txt_cuota");
 fileserver.clases.Grupo unGrupo=new fileserver.clases.Grupo();
 unGrupo.setIdGrupo(grupo.getIdGrupo());
 unGrupo.ObtenerGrupoCuota(conexion);

boolean existeUsuario=!(unGrupo.getIdGrupo()==null ||
unGrupo.getIdGrupo().equals(""));
 if(existeUsuario)
 {
  unGrupo.setEstado("1");
  unGrupo.setIdGrupo(grupo.getIdGrupo());
  unGrupo.setDescripcion(grupo.getDescripcion());
  unGrupo.setCuota(grupo.getCuota());
  unGrupo.ModificarCuota(conexion);
  // ejecutar script
  Runtime.getRuntime().exec("SAsignarQuotaGrupo
"+grupo.getDescripcion()+ " "+ (int)((new Double
(grupo.getCuota())) * 1024));
```

```
  }
  else
  {
   error="El Usuario ya existe";
  }

  if(error==null)
  {
   error="Guardado con Exito....";
   conexion.commit();
   }
   else conexion.rollback();
   pageContext.setAttribute("error", error);
   %>
  <table  width="350" border="0"align="center">
   <tr align="center">
    <td colspan="2">Cuota</td>
   </tr>
   <tr>
    <td align="justify" width="138">Descripcion:</td>
    <td width="202" align="justify"
     valign="middle">${grupo.descripcion} </td>
    </tr>
    <tr>
     <td align="justify">Id Grupo:</td>
     <td><input type="hidden" name="txt_idGrupo"
          value="${param['id_grupo']}"/>
     ${grupo.idGrupo}</td>
     </tr>
     <tr>
      <td align="justify">Cuota: </td>
      <td>  ${cuotas} <fmt:formatNumber value="$
{grupo.cuota}" pattern="0 MB"/></td>
      </tr>
      <tr>
      <td colspan="2" align="justify"><div
       align="center">${error}</div></td>
      </tr>
      <tr>
       <td colspan="2" align="justify"> </td>
      </tr>
      <tr align="center">
      <td  colspan="2"><a
href="/SACS/jasignarCuotaG.jsp">Volver</a> </td>


      </tr>
    </table>
<c:set var="method" value="define" scope="request"/>
<jsp:include page="jactualizarvista.jsp" flush="true"/>

</form>
```

1.4.2.4 jaccionCambiarClaveU.jsp

El jsp jaccionCambiarClaveU le va a permitir a un usuario cambiar su clave, que originalmente va a ser asignada por el administrador del sistema. Dicha acción se realizará de manera obligatoria cuando el usuario inicie una sesión en la aplicación por vez primera.

```
<jsp:useBean id="conexion" class="fileserver.Conexion"
scope="session"/>
 <jsp:useBean id="usuario" class="fileserver.clases.Usuario"
scope="session"/>

<c:import url="jactualizarDatos.jsp"/>
 <%

  String error=null;

  fileserver.clases.Usuario unUsuario=new
  fileserver.clases.Usuario();
  Trabajo.Conexion cone= new Trabajo.Conexion();
  Trabajo.Archivos formatos= new Trabajo.Archivos();

unUsuario.setIdUsuario((String)session.getAttribute("logon_idus
uario"));
   unUsuario.ObtenerUsuario(conexion);
   System.out.println(unUsuario.getIdUsuario() + "q");

   boolean existeUsuario=!(unUsuario.getIdUsuario()==null ||
unUsuario.getIdUsuario().equals(""));
   if(existeUsuario)
   {
     unUsuario.setEstado("1");

unUsuario.setIdUsuario((String)session.getAttribute("logon_idus
uario"));
     unUsuario.setNombre(usuario.getNombre());
     String
clave_encriptada=formatos.convierte_clave(request.getParameter(
"txt_clave"));
  cone.conectar();
  cone.mantenimiento("update fil_usuario set
clave_encriptada='"+clave_encriptada+"' where
id_usuario='"+unUsuario.getIdUsuario()+"'");
   cone.desconectar();
   unUsuario.setClave(usuario.getClave());
   unUsuario.ModificarClave(conexion);
  // ejecutar script
     Runtime.getRuntime().exec("SClave
"+unUsuario.getIdUsuario()+ " "+usuario.getClave());

    }

    if(error==null)
     {
      error="Guardado con Exito....";
      conexion.commit();
     }
     else conexion.rollback();
     pageContext.setAttribute("error", error);
    %>
```

```
<table  width="350" border="0"align="center">

<tr align="center">
 <td colspan="2">Cambiar Clave </td>
</tr>
<tr>
<td align="justify" width="280">Nombre de Usuario:</td><td
align="justify" valign="middle"> ${logon_usuario} </td>
</tr>
<tr>
<td align="justify" width="280">Usuario</td>
                                        <td> $
{logon_idusuario}</td>
</tr>
<tr>
<td  align="justify"> </td>
<td> </td>
</tr>
<tr>
<td colspan="2"  align="justify"><div align="center">$
{error}</div></td>
</tr>
<tr>
<td  align="justify" width="280"> </td>
<td> </td>
</tr>
<tr>
 <td colspan="2"  align="right"><div align="center"><a
href="/SACS/jcambiarClaveU.jsp">Volver</a> </div></td>
 </tr>
</table>
```

1.5 scripts.

1.5.1 SCrearUsuario.

Script para crear un usuario ejecutamos el comando smbldap-useradd de la herramienta smbldap-tools. Luego llamamos al script SClave enviándole parámetros como el nombre del usuario y la clave.

```
#!/bin/bash
#parametros donde:
#       $2 = Usuario
#       $3 = Clave
#            $1 = Grupo
smbldap-useradd -a -m -F "" $1
SClave $1 $2
```

1.5.2 SCrearGrupo

Script para crear un grupo de usuario ejecutamos el comando smbldap-groupadd de la herramienta smbldap-tools. Luego llamamos al script SCrearPermisos, SCrearGrpSamba enviándole parámetros como el nombre del grupo

```
#!/bin/bash
#parametros donde:
#       $1 = Grupo
#       $2 =
smbldap-groupadd $1
mkdir /home/samba/$1
SCrearPermisos $1
SCrearGrpSamba $1
```

1.5.3 SCrearPermisos

Script para crear permisos al grupo ejecutamos los comandos chown para asignar a root al grupo y comando chmod para darle los permisos a la carpeta del grupo.

```
#!/bin/bash
#parametros donde:
#       $1 = Grupo
chown -R root.$1 /home/samba/$1
chmod -R 2770 /home/samba/$1
```

1.5.4 SCrearGrpSamba

Script para crear la carpeta del grupo en el archivo smb.conf al final reseteamos el demonio de samba.

```
#!/bin/bash
#parametros donde:
#       $1 = Grupo
cd /etc/samba
echo "["$1"]" >>smb.conf
echo "comment = Archivos comunes de "$1
>>smb.conf
echo "path = /home/samba/"$1>>smb.conf
echo "valid users = @"$1>>smb.conf
echo "write list = @"$1>>smb.conf
echo "force group = "$1>>smb.conf
echo "create mode = 770">>smb.conf
echo "directory mode = 770">>smb.conf

service smb restart
```

1.5.5 SAsignarGrupo

Script para asignar un usuario a un grupo crear la carpeta del grupo en el archivo smb.conf al final reseteamos el demonio de samba.

```
#!/bin/bash
#parametros donde:
#       $1 = Usuario
#       $2 = Grupo
ln -d -s /home/samba/$2 /home/$1/$2
smbldap-groupmod -m $1 $2
```

1.5.6 SClave

Script para asignar una clave a un usuario.

```
#!/bin/bash
#parametros donde:
#       $1 = Usuario
#       $2 = Clave
smbldap-passwd $1<< EOF
$2
$2
$2
EOF
```

1.5.7 SDesabilitarUsuario

```
#!/bin/bash
#parametros donde:
#         $1 = Usuario

smbldap-usermod -L -I $1
```

1.5.8 SDesabilitarGrupo

```
#!/bin/bash
#parametros donde:
#         $1 = Grupo
chmod -R 700 /home/samba/$1
```

1.5.9 SAsignarCuotaUsuario

```
#!/bin/bash
#parametros donde:
#         $1 = Usuario
#         $2 = quota

setquota -u $1 0 $2 0 0 /home
```

1.5.10 SAsignarCuotaGrupo

```
#!/bin/bash
#parametros donde:
#         $1 = Grupo
#         $2 = quota

setquota -g $1 0 $2 0 0 /home
```

CAPITULO 2

MANUAL DE USUARIO

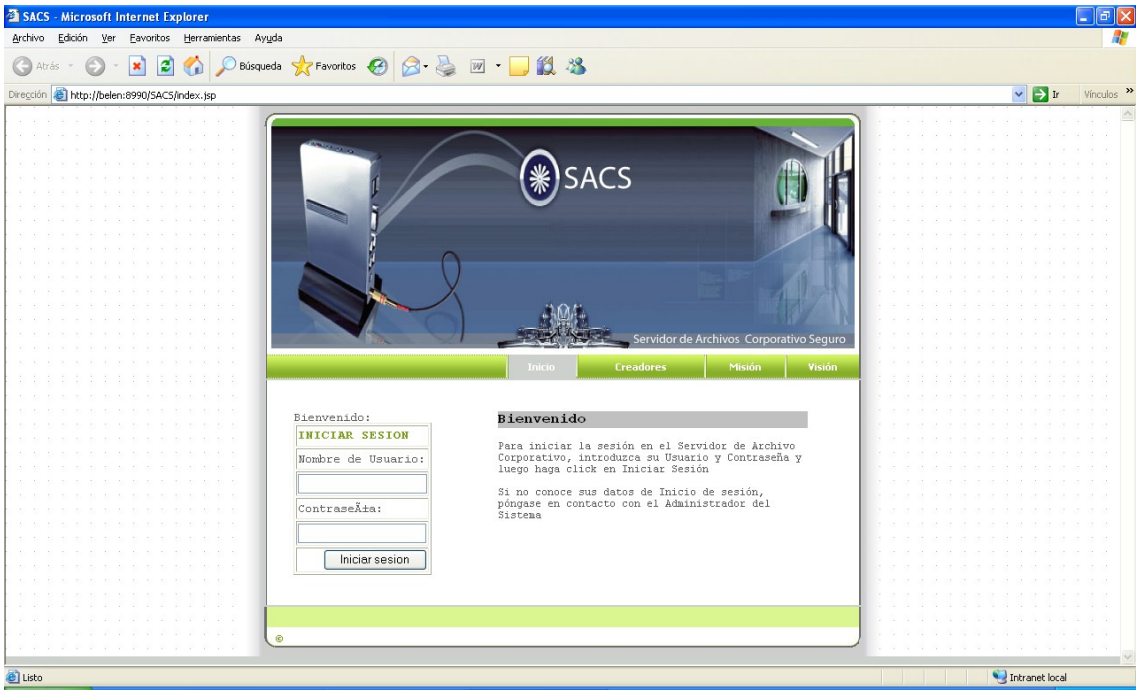
En este capitulo presentaremos las pantallas del proyecto y enseñaremos el uso de la aplicación web para administrar el Servidor de Archivos Corporativos (SACS).

2.1 Pantalla Principal

Para Ingresar a la aplicación web debemos abrir el navegador en el equipo cliente y en la barra de direcciones escribimos www.sacs.com al momento se mostrara la pantalla de login o inicio de sesión.

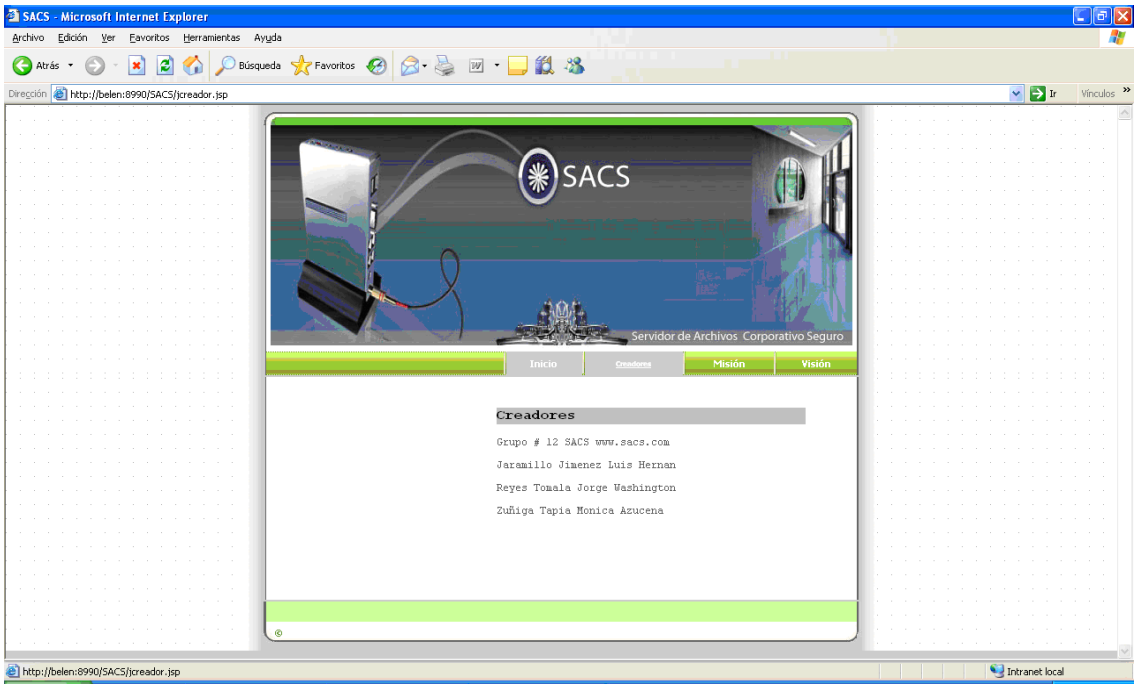
En la pantalla de inicio de sesión mostramos una Barra de Menú la cual contiene opción de Inicio, Creadores, Misión y Visión de nuestro proyecto.

Al dar clic en Inicio aparecerá una ventana de Login en donde usted deberá digitar su nombre de usuario y contraseña, luego haga clic en Iniciar Sesión, y listo usted ha ingresado al Servidor de Archivos Corporativo. Bienvenido.

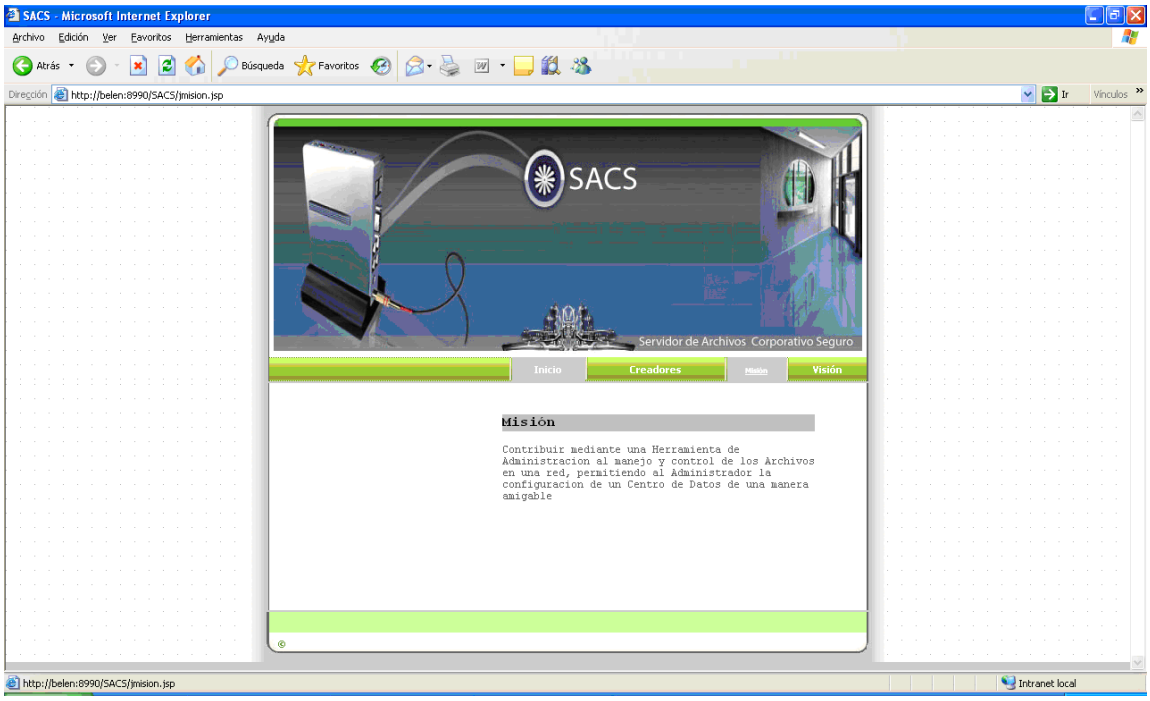


Las opciones de la barra de menú de la pantalla de inicio de sesión muestran la misión, visión, creadores del proyecto.

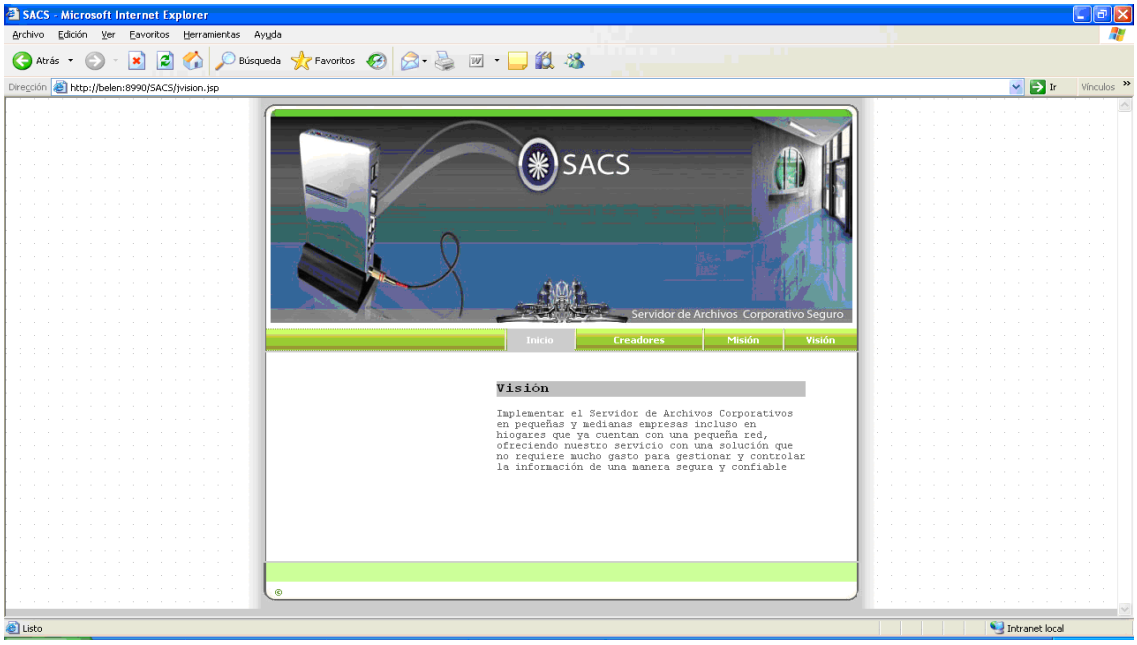
Al dar click en la opción de creadores conocerá las personas que estuvieron involucradas en el desarrollo de la aplicación Web.



Al dar clic en misión mostrara la misión del proyecto hasta donde queremos llegar y el beneficio que le representara.

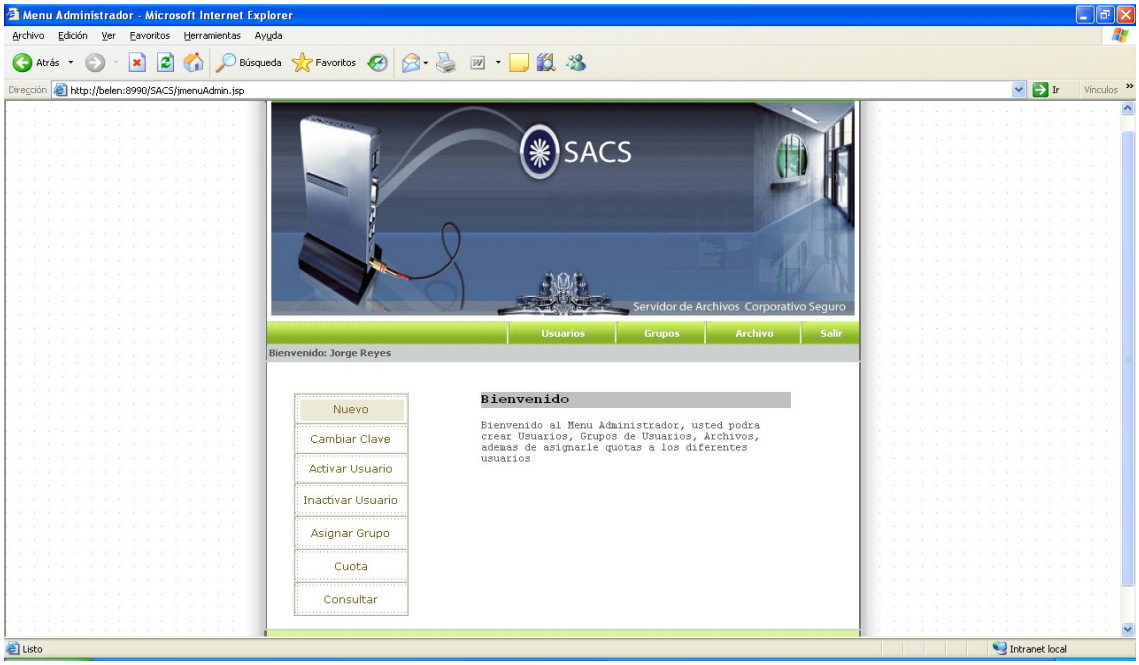

SACS - Microsoft Internet Explorer
Archivo Edición Ver Favoritos Herramientas Ayuda
Atrás Búsqueda Favoritos
Dirección http://belen:8990/SACS/jmision.jsp
Ir Vínculos
SACS
Servidor de Archivos Corporativo Seguro
Inicio
Creadores
Misión
Visión
Misión
Contribuir mediante una Herramienta de Administracion al manejo y control de los Archivos en una red, permitiendo al Administrador la configuracion de un Centro de Datos de una manera amigable
http://belen:8990/SACS/jmision.jsp
Intranet local

Si deseamos conocer la Visión del proyecto damos click en la opción visión de la barra de menú.


SACS - Microsoft Internet Explorer
Archivo Edición Ver Favoritos Herramientas Ayuda
Atrás Búsqueda Favoritos
Dirección http://belen:8990/SACS/jvision.jsp
Ir Vínculos
SACS
Servidor de Archivos Corporativo Seguro
Inicio
Creadores
Misión
Visión
Visión
Implementar el Servidor de Archivos Corporativos en pequeñas y medianas empresas incluso en hogares que ya cuentan con una pequeña red, ofreciendo nuestro servicio con una solución que no requiere mucho gasto para gestionar y controlar la información de una manera segura y confiable
Listo
Intranet local

2.2 Inicio de Sesión.

Cuando ingresamos nuestro nombre de usuario y contraseña y le damos clic en Iniciar Sesión, si el usuario existe la aplicación mostrar un menú dependiendo del perfil que tenga este usuario sea Administrador o un usuario de red, si el usuario no existe simplemente la pantalla se refrescara y borrara los datos ingresados, si el password esta mal escrito igualmente refrescará la pantalla y limpiara los datos ingresados en la caja de texto del password.



2.3 Menú Administrador

Al ingresar como Administrador la pantalla por default se ubicara en las opciones de Administración de Usuarios y mostrara en el lado derecho de la pantalla el menú para este Administrador.

2.3.1 Administración de Usuarios.

Nuevo: Crear un nuevo Usuario

Cambiar clave: Permite cambiar la clave del usuario que inicio la sesión no de otros usuarios aun sea este un Administrador.

Activar usuario: Si requerimos Activar un usuario que posteriormente a sido desactivado por razones de políticas de la empresa elegimos esta opción.

Inactivar usuario: Permite desactivar un usuario no borrarlo de la base de Datos.

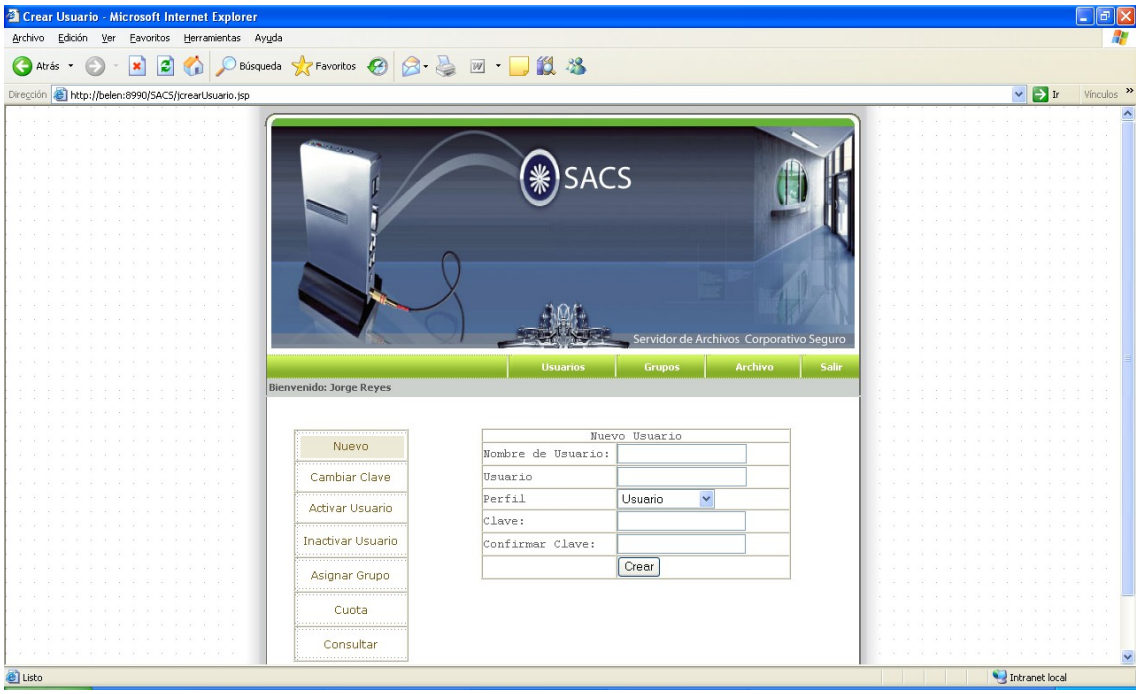
Asignar grupo: Podemos a un usuario asignarle otros grupos con esto le damos permisos para acceder a otro archivos.

Cuota: Permite asignarle espacio o cuotas de disco a un usuario.

Consultar: Muestra un reporte de los usuarios que existen en el sistema.

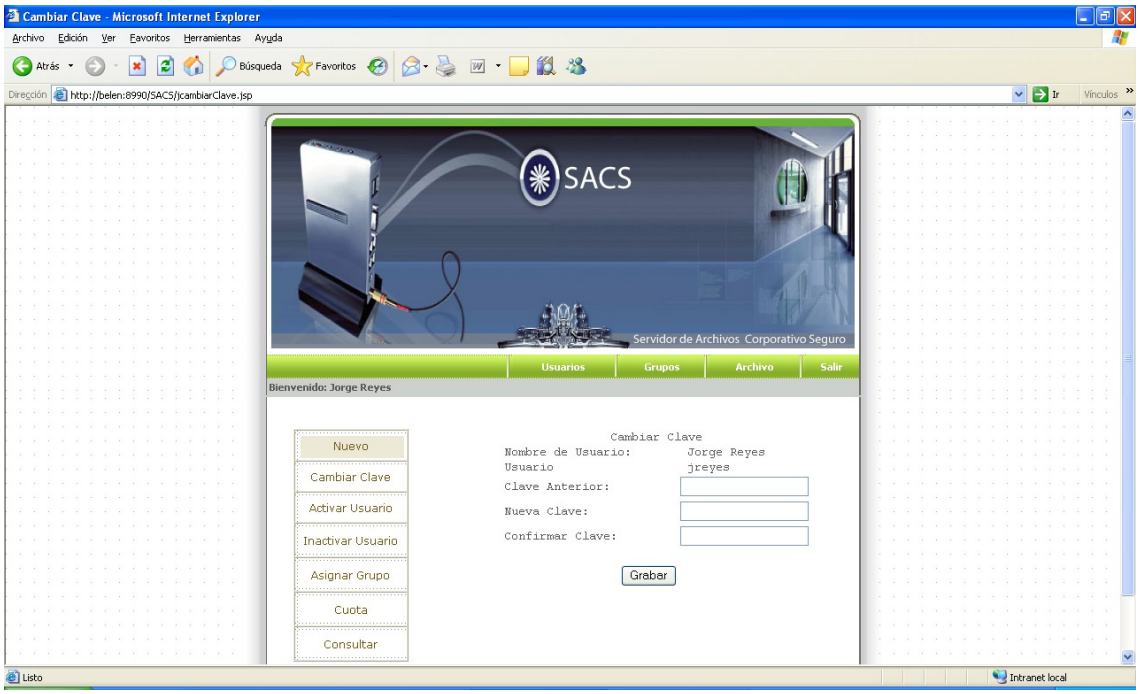
2.3.1.1 Nuevo Usuario.

Para crear un usuario damos clic en el menú de Administración de Usuarios y obtendrá una ventana de login en donde ingresamos el nombre completo de usuario y el login que desea crear, seleccionamos el perfil Administrador si tiene el privilegio de Administrador o a su vez el perfil de usuario como tal, luego ingresamos la clave y confirmarla y por ultimo dar clic en el grabar para que sus datos sean guardados en la Base de Datos.



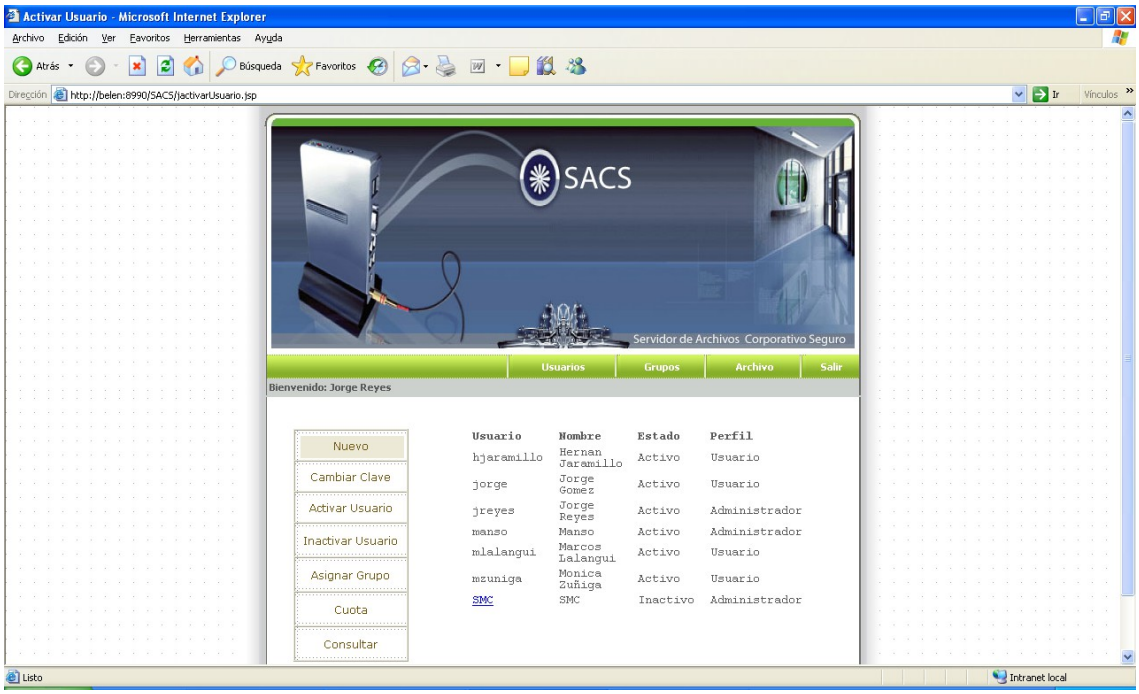
2.3.1.2 Cambiar Clave

Cuando usted ya sea usuario de nuestro sistema y desee cambiar su clave lo podrá hacer por medio de la opción cambiar clave y obtendrá una pantalla en donde para la seguridad de los usuarios la persona de desea cambiar su clave deberá ingresar su clave anterior, luego la nueva clave y por ultimo confirmarla y dar clic en grabar para que se guarden sus cambios.



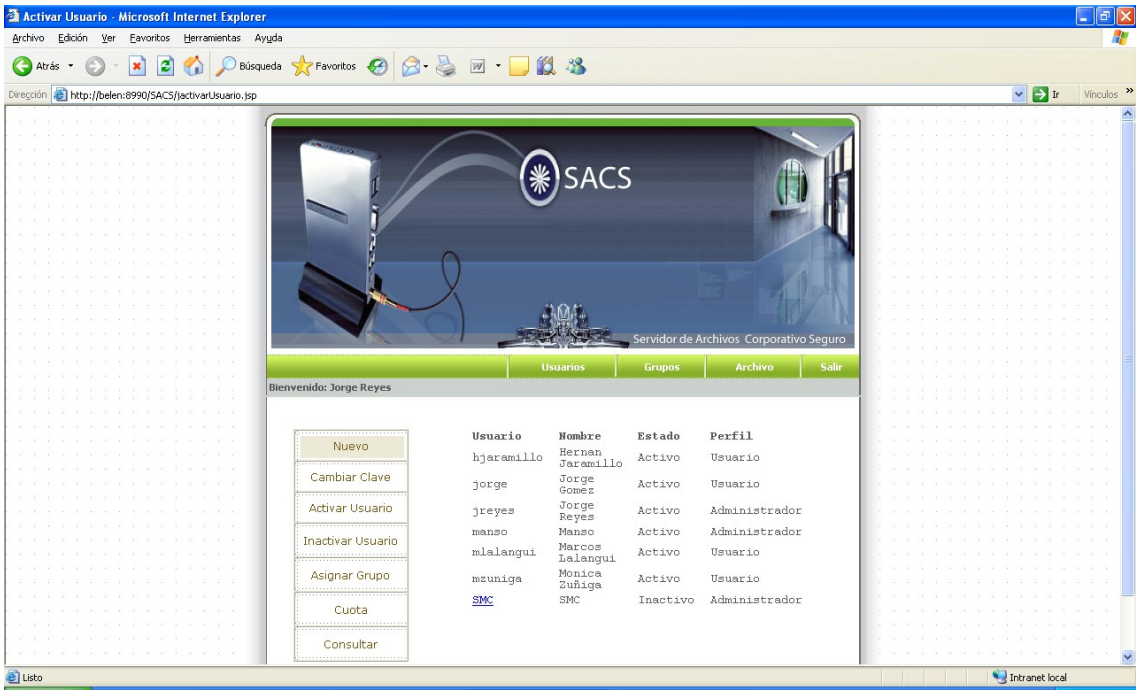
2.3.1.3 Activar Usuario

Cuando usted desee activar algún usuario que se encuentre desactivado lo hará de una manera muy fácil escogiendo la opción activar usuario este mostrara un listado de todos los usuario y su estado si esta activo o inactivo. La aplicación solamente le permitirá dar un clic en los usuarios que se encuentren en estado inactivo.



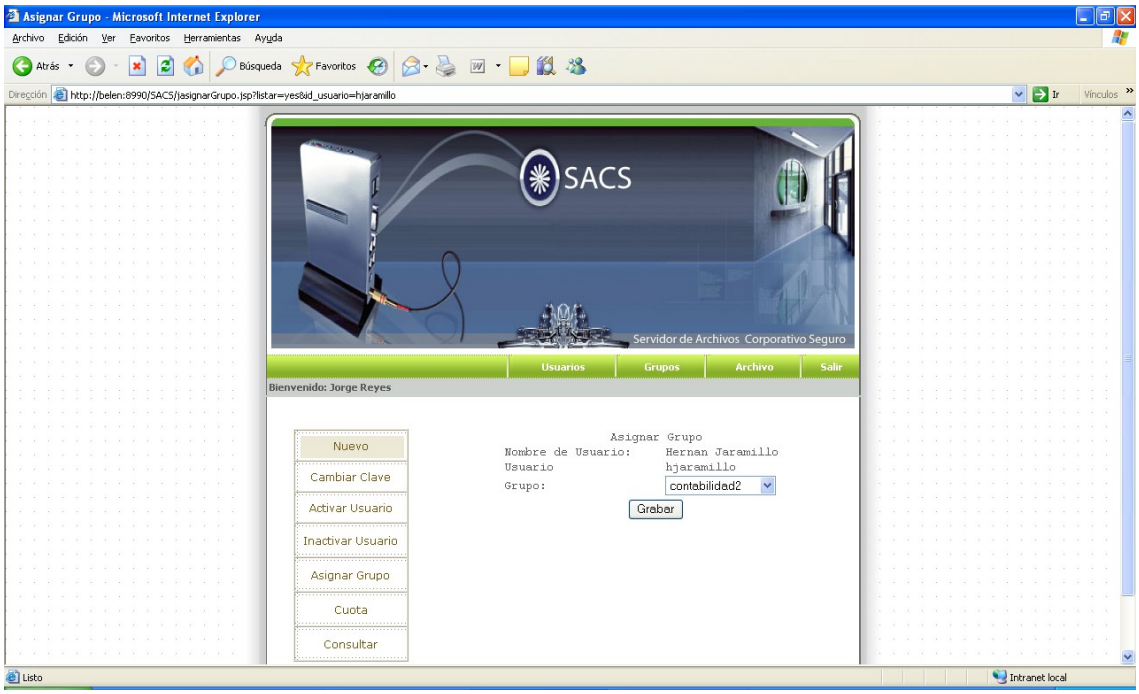
2.3.1.4 Inactivar Usuario

De la misma manera como usted activa un usuario también podrá inactivar un usuario activo dando clic en el login del usuario que se encuentre en estado activo.



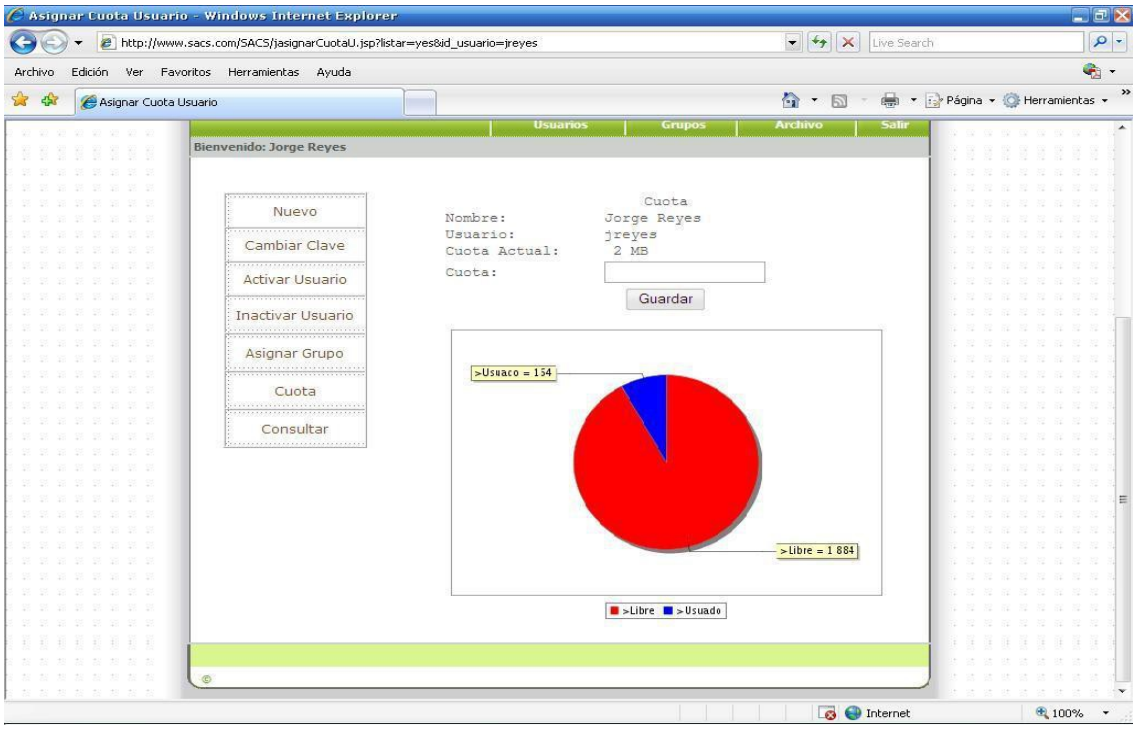
2.3.1.5 Asignar Grupo.

Luego, podemos elegir la opción Asignar Grupo en donde el sistema le permitirá asignar un determinado grupo a un usuario al dar clic la flecha hacia abajo del combobox el cual contiene todos los grupos creados en nuestro sistema y automáticamente al seleccionar un nombre de grupo usted ya pertenece a ese grupo, luego le da clic en grabar para que se guarden los datos.



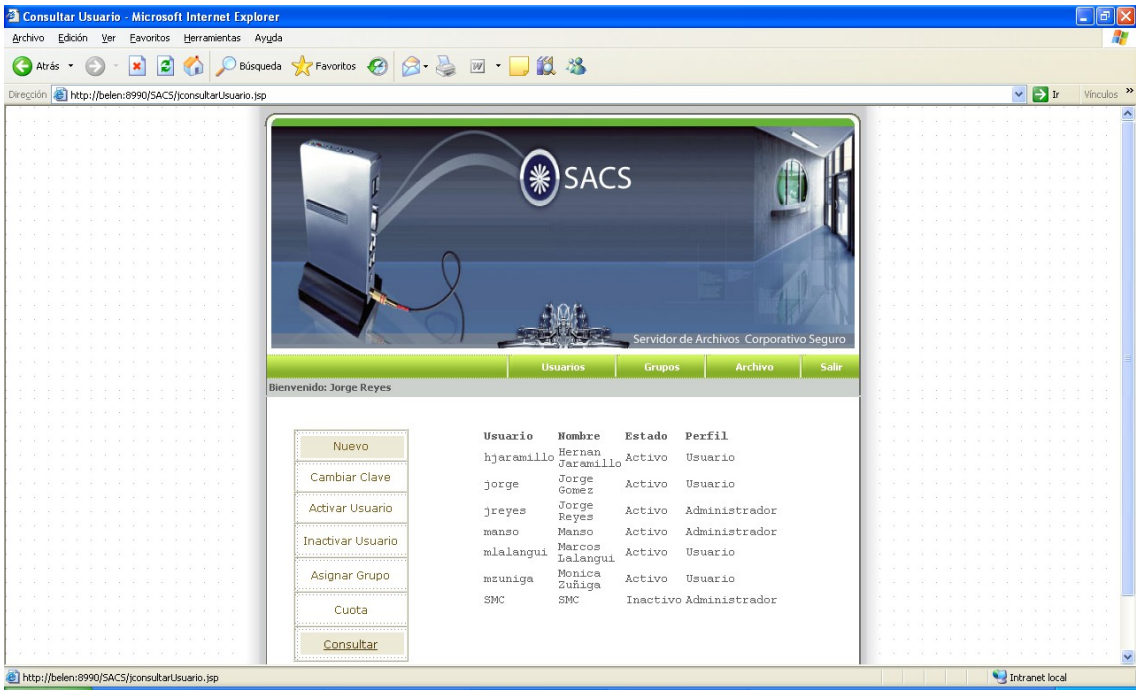
2.3.1.6 Cuota

Usted como administrador también podrá asignar quotas de disco a los diferentes usuarios, escogiendo la opción cuota por medio de la cual le podrá asignar un valor en quota para que el usuario pueda administrar su espacio en disco, de la misma manera también debe hacer clic en grabar para que todos los cambios sean guardados.



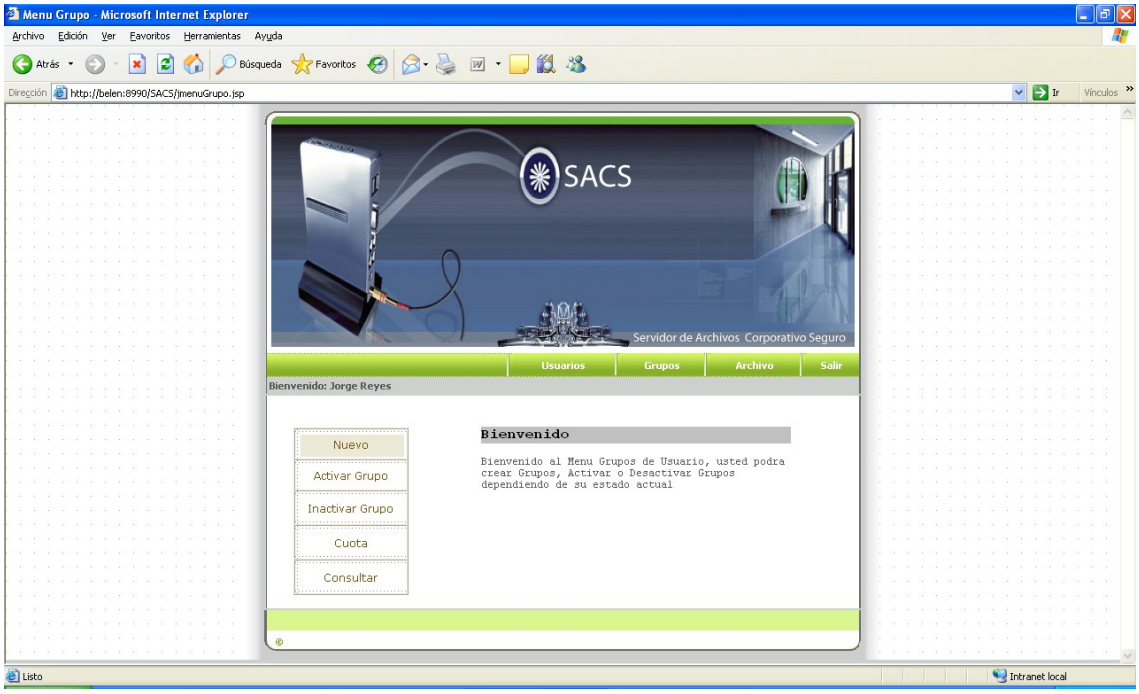
2.3.1.7 Consultar

También podrá consultar todos los usuarios que pertenecen al sistema, con sus respectivos nombres, su estado y a que perfil pertenecen.



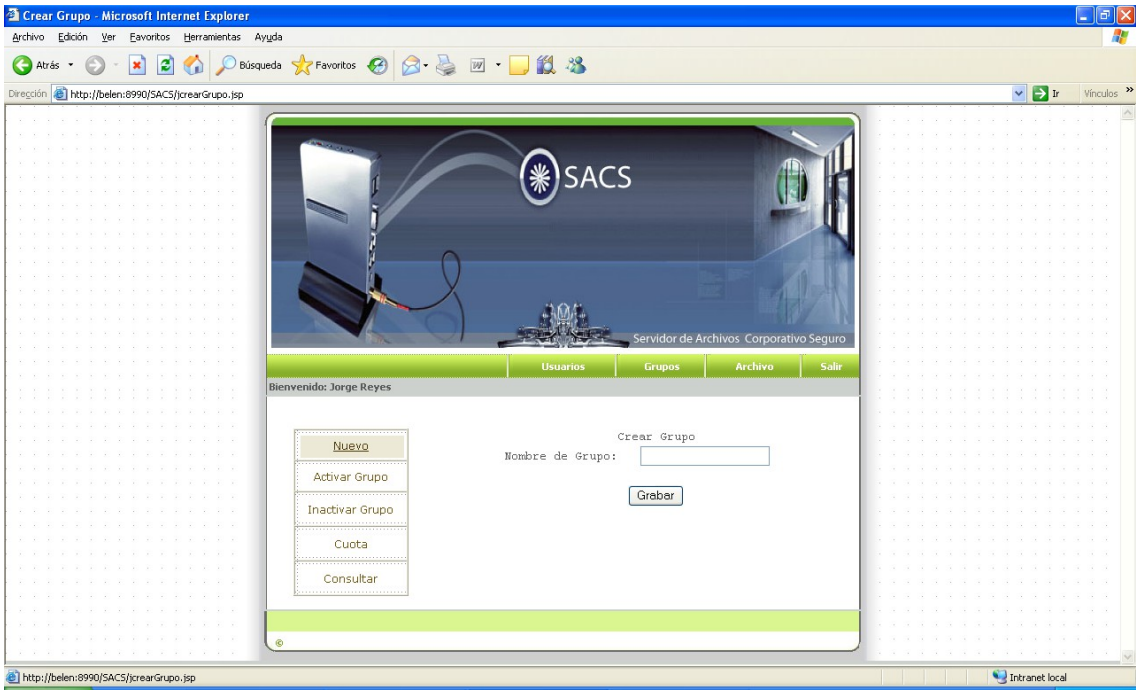
2.3.2 Administración Grupos de Usuarios

Cuando usted desee revisar la información de los grupos de usuarios de nuestro sistema podrá hacerlo escogiendo la opción Grupos, y obtendrá un pequeño menú de opciones entre ellas esta: Nuevo, activar grupo, inactivar grupo, cuota asignada al grupo y consultar todos los grupos.



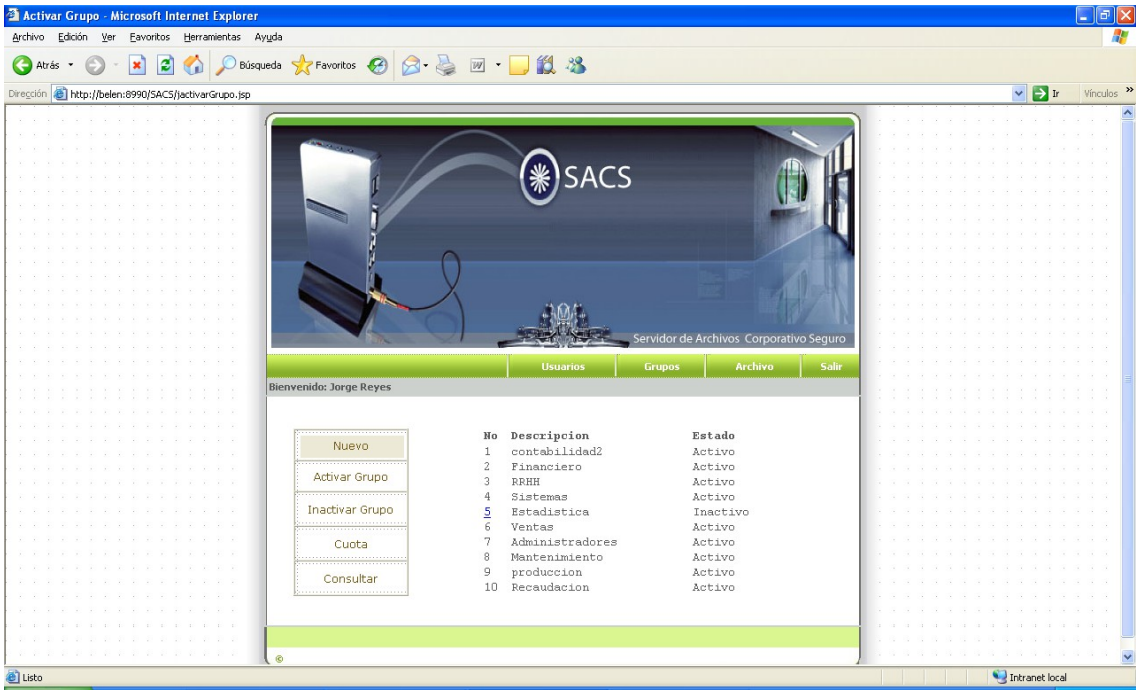
2.3.2.1 Nuevo

Si deseamos crear un nuevo grupo escogemos la opción Nuevo y automáticamente el sistema le permitirá crear un nuevo grupo y luego grabamos para que ese grupo quede registrado.



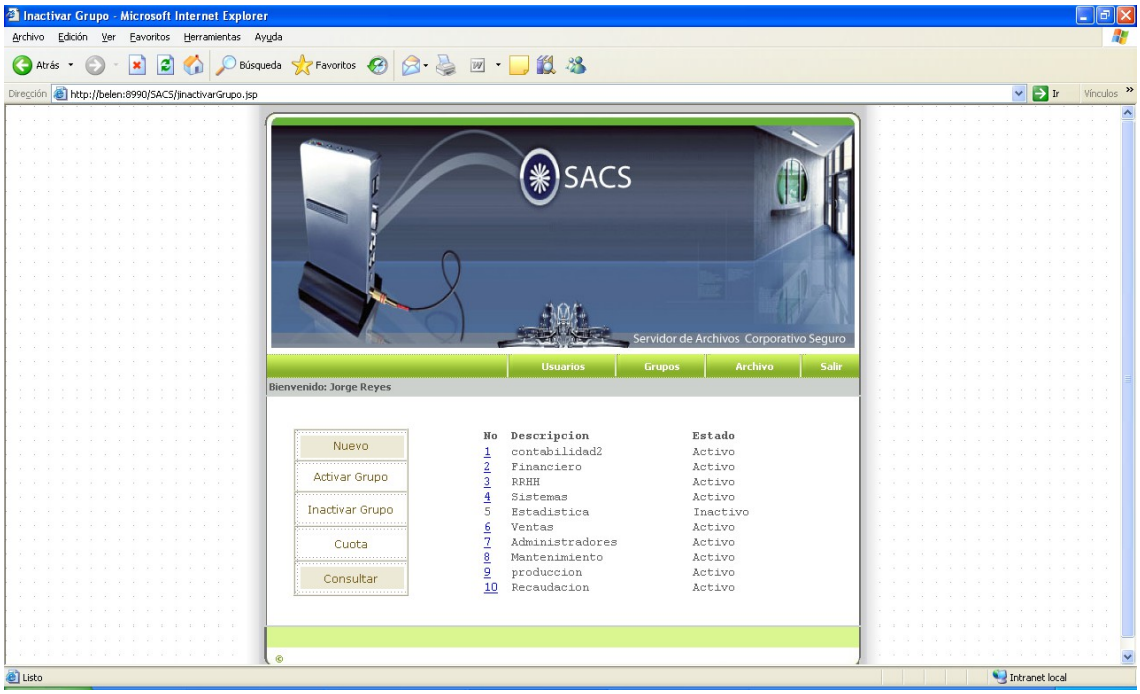
2.3.2.2 Activar Grupo

De la misma manera cuando queremos activar un grupo, verificamos que el grupo este inactivo y luego le damos clic en el numero del grupo para activarlo.



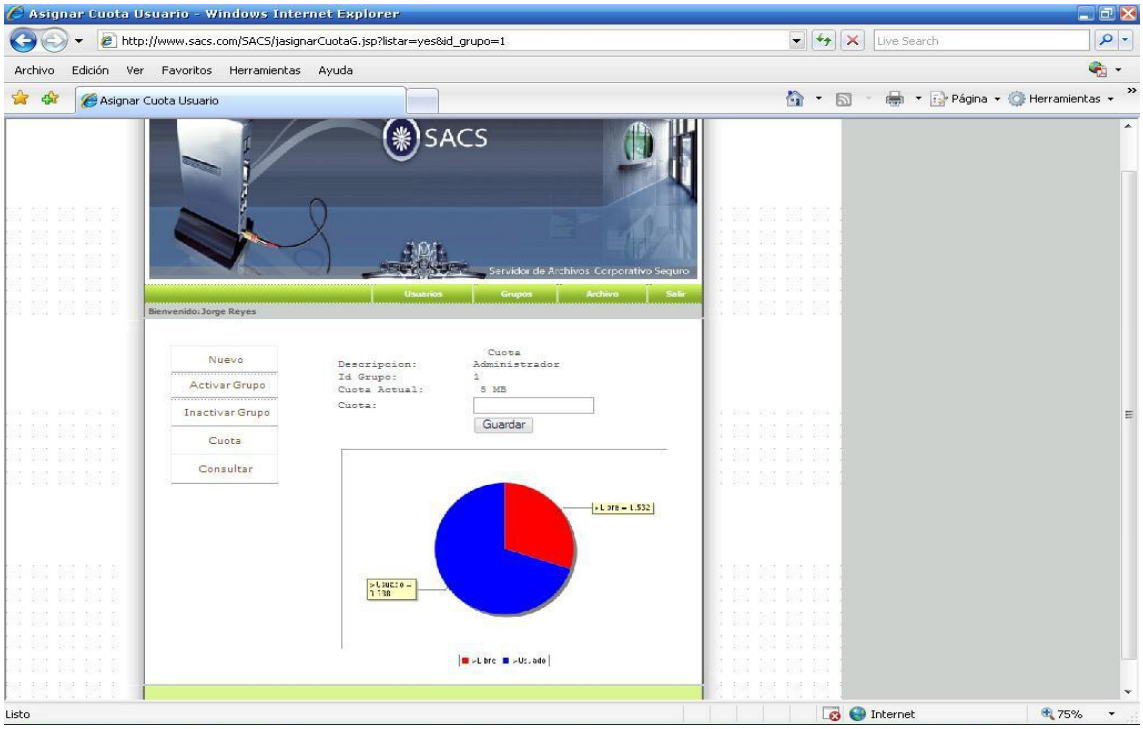
2.3.2.3 Inactivar Grupo

Igualmente cuando queremos inactivar un determinado grupo, verificamos cual es el estado del mismo y hacemos clic en el número de grupo.



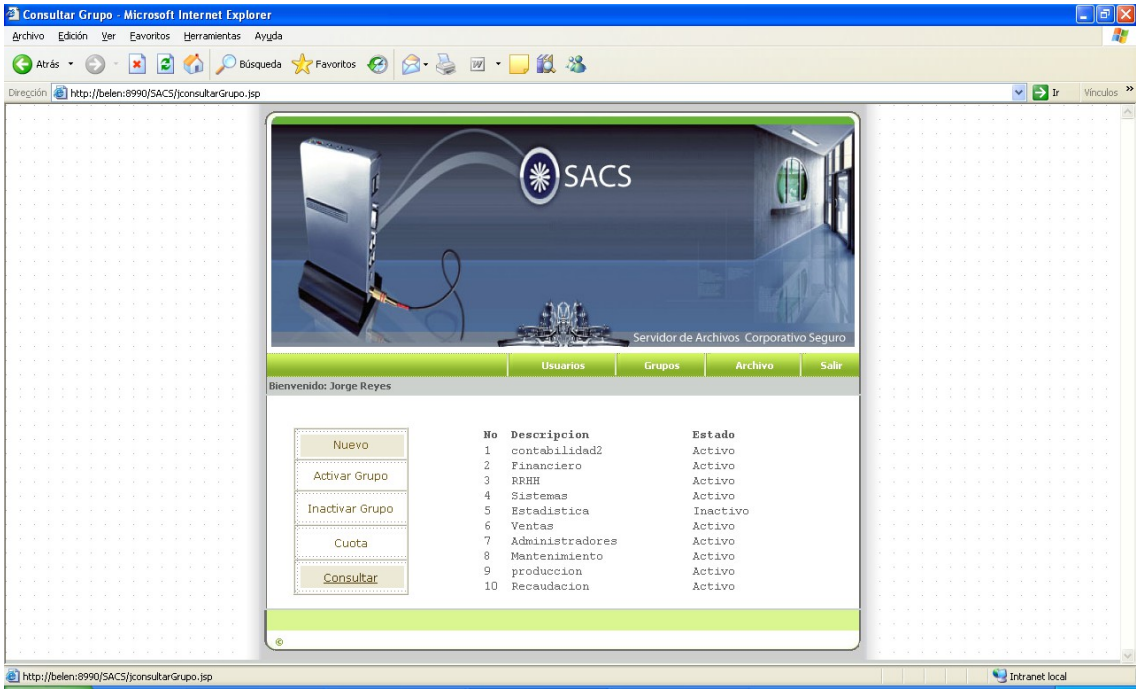
2.3.2.4 Cuotas

También este sistema le permitirá asignarle quotas de disco al grupo, es decir, un espacio en el disco duro para que los usuarios puedan manejar su información de manera responsable y utilizando eficientemente los recursos asignados, de esta manera también le permitirá al administrador llevar un mejor control del espacio total del disco duro y tomar decisiones oportunas al momento de administrar el sistema.



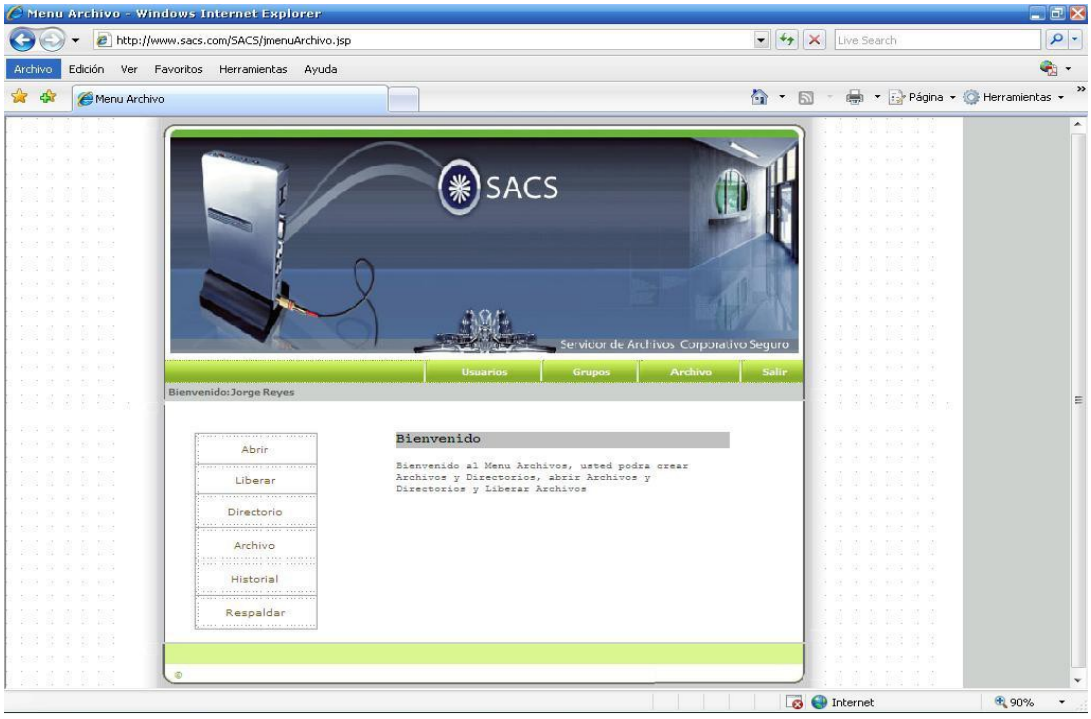
2.3.2.5 Consultar

El sistema también le permitirá realizar consultas de todos los grupos con su respectivo orden e índice y su estado.



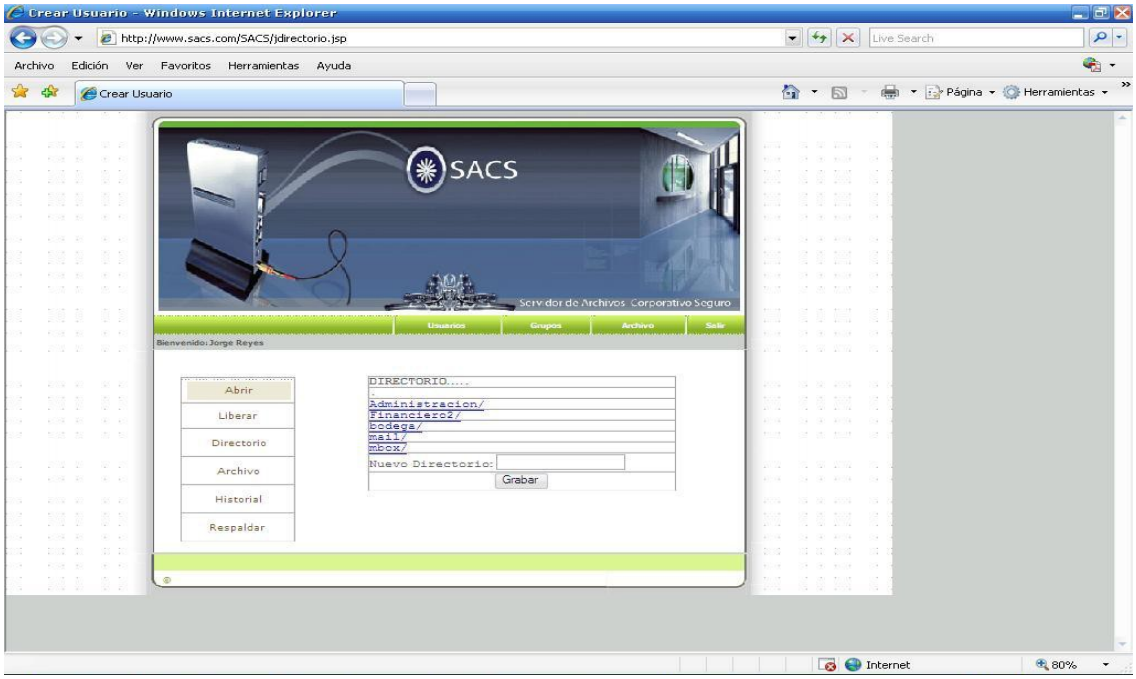
2.3.3 Administración De Archivos.

Si escogemos la opción Archivo podrá visualizar el siguiente menú que contiene las opciones de Abrir, Liberar, Directorio, Archivo, Historial y Respaldar.



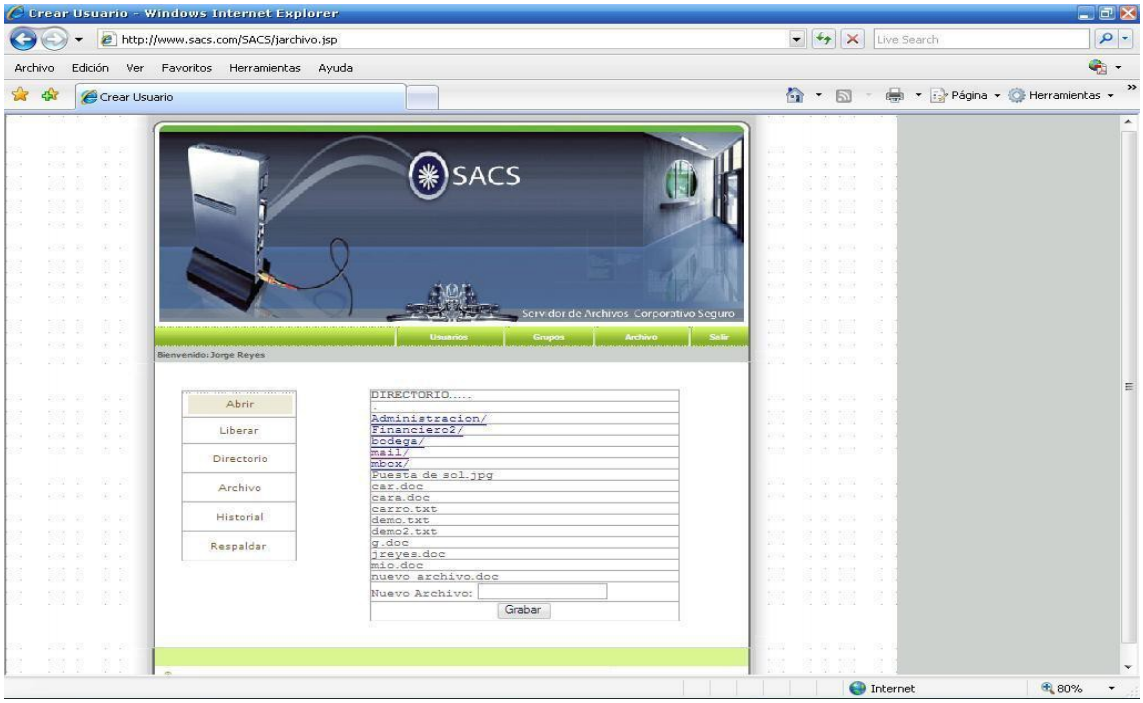
2.3.3.1 Abrir Directorio

A continuación detallamos la opción abrir, la cual le permitirá crear un nuevo directorio, grabar datos en el para que estos a su vez se alojen en el directorio home de cada usuario y resida en el Servidor y/o tener acceso a una lista directorios ya creados, haciendo clic en el nombre del directorio, siempre y cuando usted tenga el permiso respectivo para consultar la información.



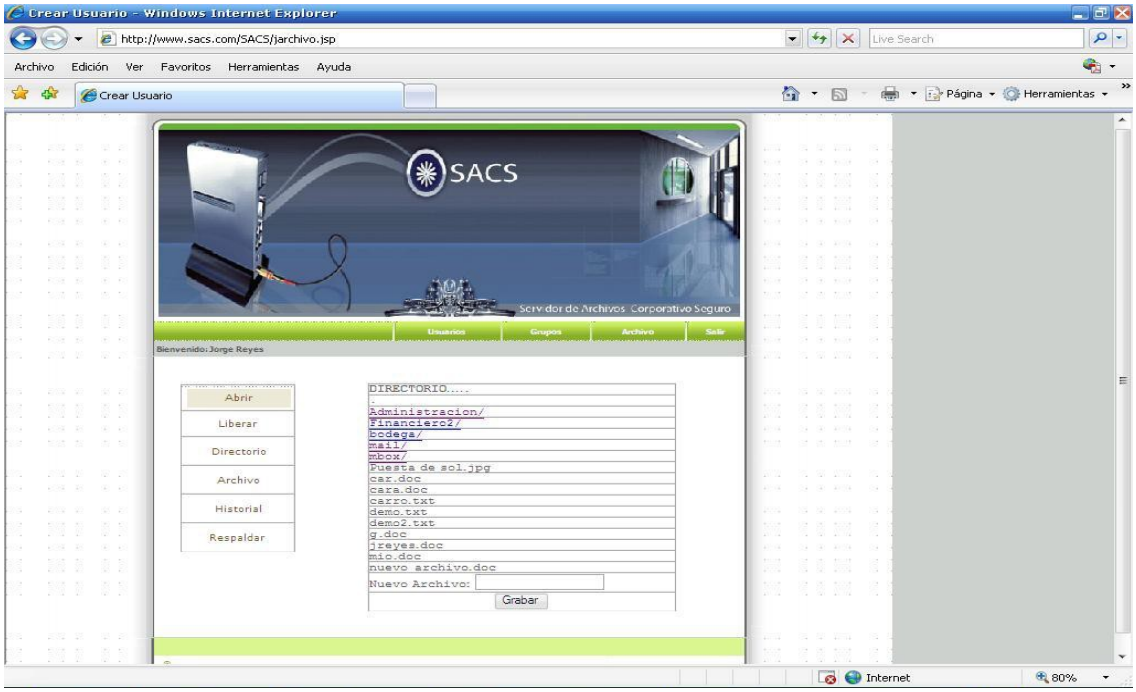
2.3.3.2 Abrir Archivo

Esta opción permitirá crear un nuevo archivo con la opción de grabar datos en el para que estos a su vez se alojen en el directorio home de cada usuario y residan en el servidor, el administrador también tendrá acceso a una lista de archivos, haciendo clic en el nombre del archivo.



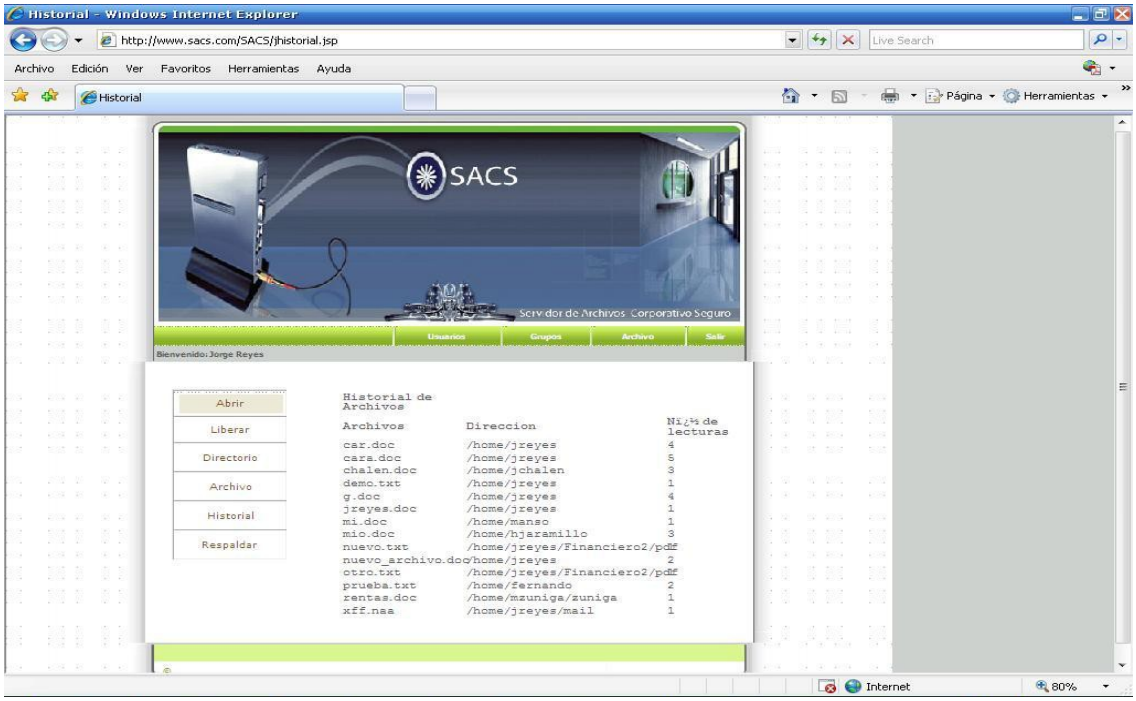
2.3.3.3 Liberar

Podrá liberar un archivo o directorio cuando usted termine de hacer uso de el, para que otro usuario de la red también lo pueda abrir.



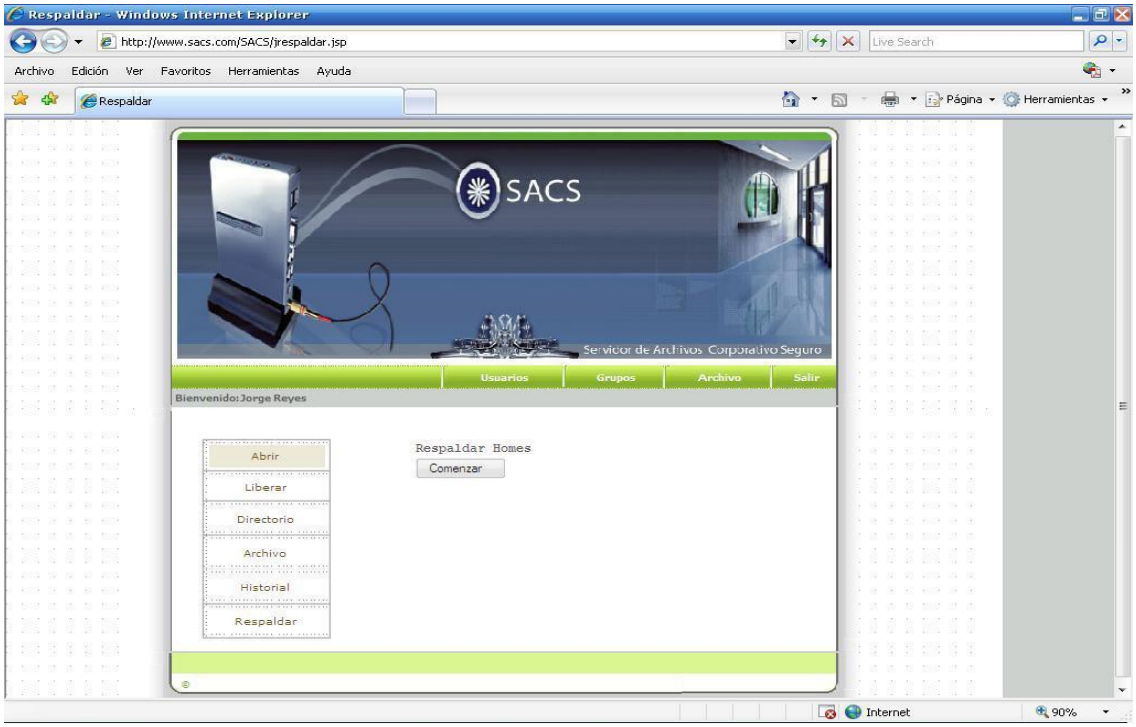
2.3.3.4 Historial

Esta opción le permitirá al administrador visualizar el historial de todos los archivos del usuario, en que dilección están ubicados y cuantas veces han sido consultados.



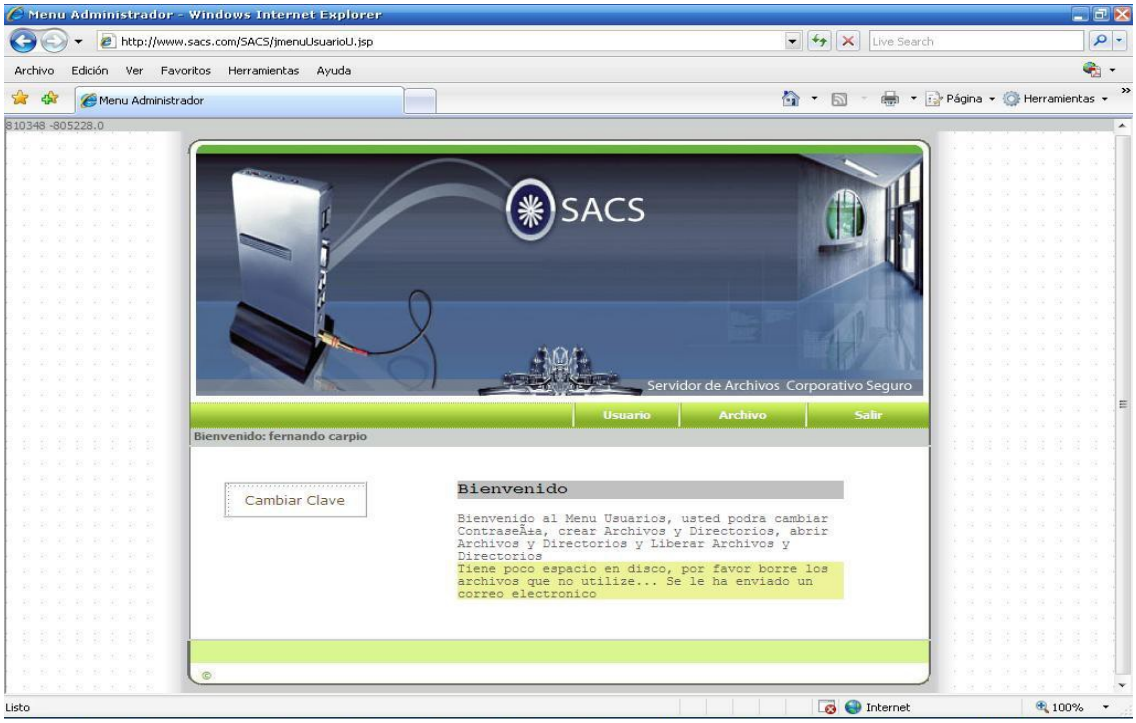
2.3.3.5 Respaldar

Esta opción le permite al administrador respaldar el contenido del directorio Home del usuario.



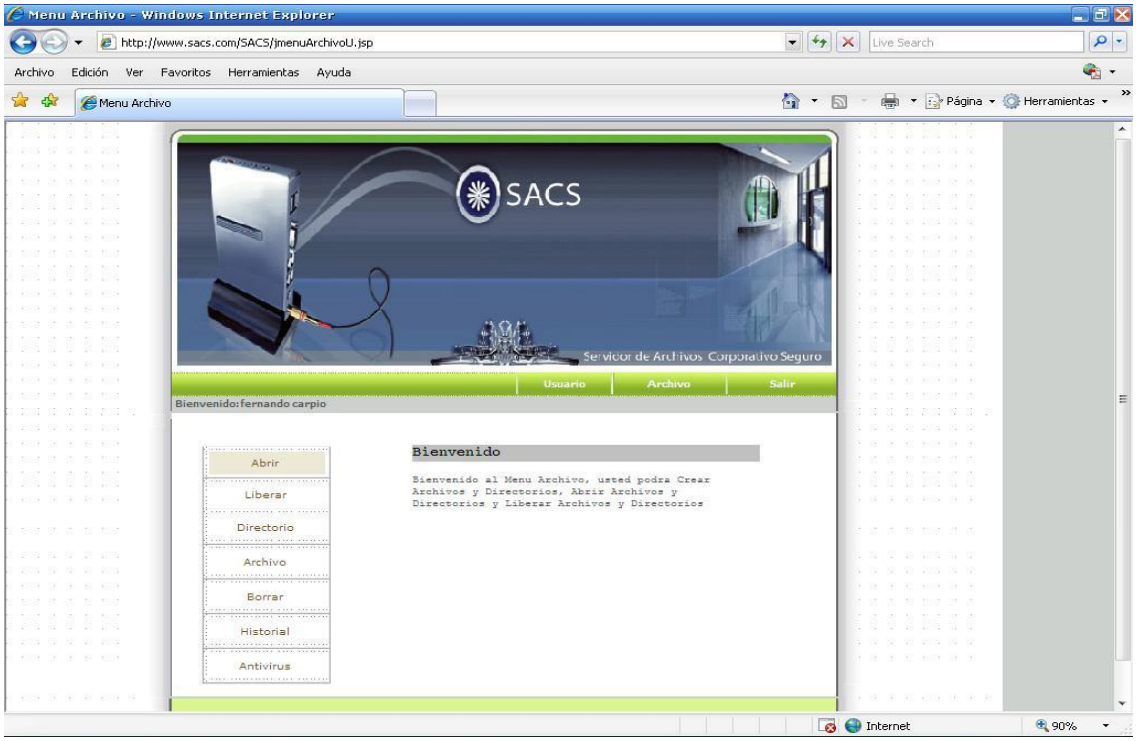
2.4 Menú Usuario

Este menú le permitirá al usuario cambiar su clave personal y manipular el Menú Archivos con las opciones de Crear Archivos y Directorios, Abrir Archivos y Directorios, Liberar Archivos y Directorios, Borrar Archivos, Historial, Antivirus, además cada usuario recibirá mensajes de correo electrónico cuando su cuota de usuario este por exceder el limite asignado.


Menu Administrador - Windows Internet Explorer
http://www.sacs.com/SACS/jmenuUsuarioU.jsp
Archivo Edición Ver Favoritos Herramientas Ayuda
Menu Administrador
SACS
Servidor de Archivos Corporativo Seguro
Usuario
Archivo
Salir
Bienvenido: fernando carpio
Cambiar Clave
Bienvenido
Bienvenido al Menu Usuarios, usted podra cambiar ContraseÃ±a, crear Archivos y Directorios, abrir Archivos y Directorios y Liberar Archivos y Directorios
Tiene poco espacio en disco, por favor borre los archivos que no utilize... Se le ha enviado un correo electronico
Listo
Internet
100%

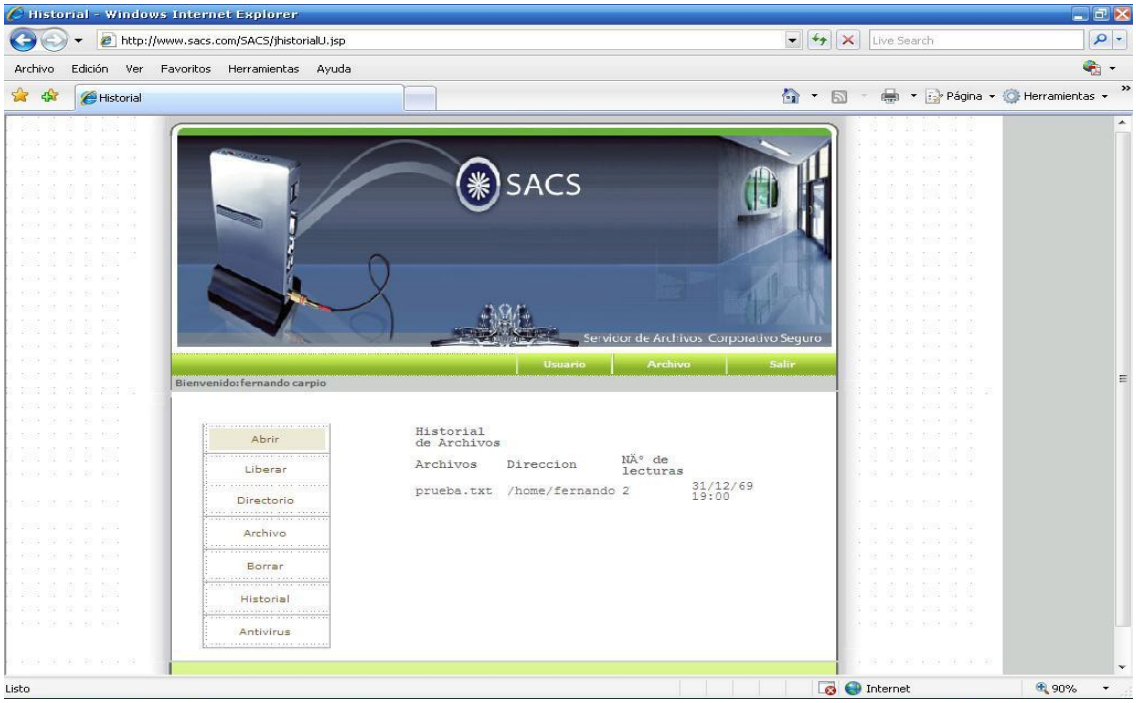
2.4.1 Menú Archivos

Menú Archivos tiene las siguientes opciones: Crear Archivos y Directorios, Abrir Archivos y Directorios, Liberar en Archivos y Directorios, Borrar Archivos, Visualizar el Historial de todos los Archivos del Usuario, utilizar un Antivirus para que pueda salvaguardar su información.



2.4.1.1 Historial

Esta opción le mostrará al usuario el historial de todos sus archivos para que pueda consultar cuál es el tipo de archivo, en que ruta esta ubicado, cuantas veces ha sido consultado y la fecha y hora de la última consulta.



2.4.1.2 Antivirus

Esta opción realiza el escaneo de todos los archivos del usuario ubicados en el directorio home por medio del Antivirus F-prot para Linux el cual después de realizar el escaneo de los archivos retorna el total de los archivos, los objetos escaneados, los archivos infectados y los archivos desinfectados.